NetScaler 10.1

About the NetScaler 10.1 Release

Jan 01, 2014

NetScaler 10.1 releases are categorized into the following:

Main releases - Includes the first main NetScaler 10.1 build and subsequent builds which typically include minor changes and fixes to issues that existed in previous builds. In some instances, main releases might also include major enhancements.

Release naming format: NetScaler <ver> Build <major_ build>.<minor_build>.

Enhancement releases - Includes builds that provide major enhancements to a previously released NetScaler main release.

Release naming format: NetScaler <ver> Build <main_release><minor_build>.e.

For example, a release that adds an enhancement to NetScaler 10.1 Build 123.11 would be: NetScaler 10.1 Build 123.11xx.e.

> **Important**
>
> Unless explicitly specified, these enhancements are not available in main releases.

Main Releases

Aug 10, 2015

The release notes describe the changes or enhancements, fixed issues, and known issues in Build 133.9. The list of known issues is cumulative, that is, it includes issues that are newly found in this build and also issues from previous builds.

Fixed Issues - The issues addressed in Build 133.9.

Known Issues - The issues that exist in Build 133.9.

What's New in Previous 10.1 Builds - The enhancements and changes that were available in NetScaler 10.1 releases prior to Build 133.9. The build number provided below the issue description indicates the build in which this enhancement or change was provided.

Fixed Issues in Previous 10.1 Builds - The issues that were addressed in NetScaler 10.1 releases prior to Build 133.9. The build number provided below the issue description indicates the build in which this issue was addressed.

Note

- This build includes fixes for 7 issues that were known issues in the previous build of the NetScaler 10.1 release.
- The [# XXXXXX] labels under the issue descriptions are internal tracking IDs used by the NetScaler team.
- These release notes do not document security related fixes. For a list of security related fixes and advisories, see the Citrix security bulletin.

Fixed Issues

Oct 10, 2015

The issues addressed in Build 133.9.

AAA-TM | Acceleration | AppFlow | Application Firewall | Configuration Utility | DNS | GSLB | Load Balancing | NITRO API | NetScaler Gateway | NetScaler Insight Center | NetScaler SDX Appliance | Networking | Platform | SSL | System

---

- With LDAP authentication, users can experience authentication failures even when they provide valid credentials. This issue occurs when the authentication subsystem is low on memory after a large number of invalid authentication attempts.

  [#534280]

- The classic-policy expression used by the default acceleration policy fails to identify an Internet Explorer browser whose signature does not comply with the IE user-agent string standards.

  [#535130]

- The NetScaler appliance might fail if, while AppFlow for ICA is enabled, a network glitch disrupts the Citrix Receiver connection and Receiver attempts to reconnect.

  [#531017, 532712, 542000, 544421, 547598, 547984, 548297, 548749, 548771, 549044, 549370, 549511, 567534, 578548]

- The Citrix application firewall silently resets the connection when it receives a malformed or invalid request. With this fix, the application firewall logs such events.

  [#577742]

- If a large number of long standing sessions expire and are freed during application firewall processing, a tight-loop condition might occur, causing the NetScaler appliance to fail.

  [#550657]

- If, when processing a form for response-side security check inspection, the application firewall resets a connection, the partially parsed form is not freed. The result is a memory leak. With this fix, the memory allocated to the partially parsed forms is freed when a connection is reset.

  [#572637, 581520]

- When you update the application firewall signatures from the NetScaler command line, you must update the default signatures first, and then issue additional update commands to update each custom signatures file that is based on the

default signatures. If you do not update the default signatures first, a version mismatch error prevents updating of the custom signatures files.

For example, if you had two sets of custom signatures, named "custom_signatures" and "custom_signatures_2", that were based on copies of the default signatures file, you would update the signatures on your NetScaler ADC by issuing the following commands:

> update appfw signatures "*Default Signatures"

> update appfw signatures "custom_signatures"

> update appfw signatures "custom_signatures_2"

[#399596]

- During an application firewall security check inspection, a compressed response from the server might trigger a violation if the XML format check is enabled. With this fix, the Accept-Encoding request header is removed when the XML protections are enabled. If content compression is enabled on the server, the XML check inspection is bypassed when the server sends a compressed response.

  [#580273]

- If you specify the service type as DNS and select the DNS64 and ByPassAAA check boxes, and later navigate to some other service type (for example HTTP), the checkboxes are grayed out, because they do not apply to (for example) an HTTP service, but they are not cleared. That is, DNS64 and ByPassAAA are disabled but not set to the default value.

  [#538163]

- The system backup and restore functionality is not available on the Cisco NetScaler GUI.

  [#553373]

- Non-standard query packets are altered before they are forwarded to back-end servers, which causes the server to respond with a "FORMAT error" message.

  [#559064]

- Loading a new location file that has a coordinate outside the correct range (-90 to +90 latitude or -180 to +180 longitude) can cause the appliance to fail.

  Recommendation: After loading any location file, use the command, "show locationparameters" to get a summary of the coordinates loaded and any parsing errors. The specific problems are reported in /var/log/ns.log.

  [#550294]

- If the "Invalid argument error" message appears intermittently in nsmund.log, treat it as a false positive. The error

appears because a scenario was not handled correctly. However, if this message appears in the log every time a particular script runs, there is an issue with the arguments that are passed to the script.

[#568719]

- In a RADIUS load balancing setup, if Use Source IP (USIP) is configured on the RADIUS services, the server side connections are not reused, and requests are dropped.

  [#574120, 534888]

- In a RADIUS load balancing setup, requests might be dropped because the memory for the session entries is not freed until the idle timeout expires even though the transaction completed earlier.

  [#573155]

- If you configure cookie persistence and custom cookie on a virtual server, and later change the name or IP address of the virtual server, persistence is not honored.

  [#524079, 559022]

- If an SSL monitor is bound to a domain-based service that is configured with non-default SSL settings, the monitor might not show the service as UP.

  [#575171, 576012]

- In a high availability setup, if custom cookie persistence is configured on a virtual server, part of the secondary node's configuration might not be synchronized with the primary after a failover occurs.

  [#552799, 552607]

- If the load balancing (LB) feature is not licensed, and you try to enable an LB virtual server, an error message appears.

  [#466094, 534755]

- **Viewing the Statistics of Services and Service Groups that are Bound to a Load Balancing Virtual Server**

  You can now view the statistics of services and service groups that are bound to a load balancing virtual server by using the following URL:

  http://<netscaler-ip-address>/nitro/v1/stat/lbvserver/<name>?statbindings=yes

  You cannot view these details by using the "http://<netscaler-ip-address>/nitro/v1/stat/lbvserver/<name>" URL which only gives the statistics of the load balancing virtual server.

  [#241950, 244603, 523907, 534804, 538057]

- **Applicable only for Mac VPN clients**

  Chrome is phasing out NPAPI support. From Chrome version 42+ all NPAPI plugins will appear as if they are not installed. This will affect all existing customers. Affected customers will see a download prompt even though the VPN plugin is

installed.

Workaround: Google has announced that Chrome will stop supporting NPAPI completely in version 45.

Until then, you can enable NPAPI as follows:

1) In the Chrome URL bar, type:

Chrome://flags

2) Enable the "Enable NPAPI" option.

3) Restart Chrome.

For more information about NPAPI deprecation, see https://support.google.com/chrome/answer/6213033?hl=en

[#572447, 574353, 575609]

- In a double hop deployment, STA server status on hop 1 does not go down when double hop is disabled in hop2, and the user still launch ICA apps.

  [#539743]

- Internet access fails intermittently when connecting to a NetScaler Gateway with Split Tunnel On using a Windows machine.

  [#572709]

- NetScaler Gateway EPA and VPN plugins don't get triggered on latest chrome browser. Chrome browser shows download prompt even after installation.

  [#570493]

- NetScaler Gateway now supports Windows 10.

  [#579428]

- The RSA Pin change fails if RSA radius servers are load-balanced with the RADIUS type protocol service. The workaround is to change the load-balance protocol service type to UDP or ANY.

  [#534888, 528950, 542189]

- NetScaler appliance fails because of incorrect handling of HDX Insight's internal data structures. This may happen when HDX Insight skips parsing ICA data in certain error scenarios.

  [#559043, 553185, 585888, 588152]

- NetScaler appliance fails because of incorrect handling of HDX Insight's internal data structures. This may happen when HDX Insight skips parsing ICA data in certain error scenarios.

  [#551081, 580514, 589856]

- On an SDX appliance, the Management Service may lose connectivity. The issue is seen only with Management Service which is in the UP state for many days, minimum being 277 days.

  [#444854, 487984, 496194, 506802, 547064, 547571, 549842]

- If you have created channels on NetScaler SDX Appliance, the Management Service statistics process, svm_stat, may fail in some cases.

  [#570006]

- Upgrading XenServer on a NetScaler SDX appliance to Revision 1 of XenServer 6.1 causes a loss of information about memory and CPU settings assigned to the control domain. As a result, a subsequent attempt to upgrade to XenServer 6.5 fails.

  [#578680]

- A PBR6 rule might not get evaluated if you set the operator option to NEQ (!=) for source and destination IPv6 addresses.

  [#575906]

- In a high availability configuration, if you remove an ACL rule from the primary node and modify another ACL rule on the primary node, but you do not apply the ACLs on the primary node before forcing synchronization on the secondary node, the secondary node might become unresponsive.

  [#576810, 545920]

- An ACL6 rule might not get evaluated if you set the operator option to NEQ (!=) for source and destination IPv6 addresses.

  [#573516]

- An attempt to access the configuration utility might fail if the logon address is an IPv6 address.

  [#553588]

- If you have configured an INAT rule in which the private IP address is set to a virtual IP address, the rule is removed after you restart the NetScaler appliance.

  [#556632]

- TFTP monitor probes might fail with the error "Probe Timed out."

  [#578663]

- If the IPv6 routes change, the IPv6-Ipv6 tunnel's encapsulation IP addressess are not obtained based on the latest route information. As a result, the tunnels use old encapsulation IP addresses to encapsulate packets.

  [#564252]

- The MPX 25100T and MPX 25160T platforms are now supported in this release. For more information about these platforms, see http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-platforms-con/ns-hardware-25100T-25160T-ref.html.

  [#486703, 495591, 552218]

- You can now upgrade the Lights Out Management (LOM) firmware directly from the host, without configuring the LOM port on the network.

  [#430733, 542439]

- If you have a cluster setup of NetScaler MPX 8005/8015/8200/8400/8600/8800 appliances, time synchronization among the cluster nodes might fail.

  [#356564, 566811]

- If application data is received during an SSL renegotiation handshake, the appliance sends a RST flag.

  [#542034]

- In some cases, when client authentication is enabled, incorrect data form a client leads to a memory leak on the NetScaler appliance. If a large number of clients send incorrect data, the appliance fails.

  [#570754]

- In a NetScaler cluster setup, if you add a certificate-key pair with a subject name greater than 64 characters, and later add more certificate-key files, the addition fails with the "No such certificate file exists" error even though the certificate-key pair key file is present on all cluster nodes.

  [#554917]

- If you execute NTP commands, such as enable ntp sync and show ntp status, the NetScaler appliance might become unresponsive because of a memory leak.

  [#529787, 574866, 581849]

- NTP Version Update

  In NetScaler release 11, the NTP version has been updated from 4.2.6p3 to 4.2.8p2.

  If you upgrade your NetScaler appliance from any earlier release to release 11, the NTP configuration is automatically upgraded with additional security policies. For more information about configuring an NTP server, see http://docs.citrix.com/en-us/netscaler/11/system/basic-operations/configuring-clock-sychronization.html.

  [#440375, 440591]

- If you enable the snmp alarm SERVICEGROUP-MEMBER-MAXCLIENTS, varbinds such as svcGrpMemberName, svcGrpMemberEstablishedConn, alarmHighThreshold, svcGrpMemberFullName, and sysIpAddress might be missing from the alert.

[#578673]

Known Issues

Oct 10, 2015

The issues that exist in Build 133.9.

AAA-TM | AppFlow | Application Firewall | CloudBridge Connector | Cluster | Command Line Interface | Configuration Utility | Content Switching | Content Switching/Load Balancing | DNS | DataStream | GSLB | Graphical user Interface | High Availability | Integrated Caching | Load Balancing | NITRO API | NS-AAA | NS-Gateway | NetScaler 1000V | NetScaler Gateway | NetScaler Insight Center | NetScaler SDX Appliance | NetScaler VPX Appliance | Networking | Platform | Policies | Reporting | SSL | System | User Interface | Web Interface | XML API

- The NetScaler implementation of Kerberos does not fully implement the ktutil functionality. While this does not affect Kerberos authentication, it restricts some administrative tasks, such as the ability to merge keytab files.

  [#551091]

- Forms-based single sign-on does not work if the form is customized to include Javascript.

  [#565740]

- The NetScaler ADC AAA-TM user interface has a timeout of 20 seconds. If authentication through an external authentication server takes more than 20 seconds, the following message appears in the logs: "libaaa recv failed." This message does not indicate authentication failure or any other problem that affects users. It can safely be ignored.

  [#437454]

- The rule (expression) in a AAA-TM policy can be from one to 1434 characters in length. If you enter a longer rule, AAA-TM displays an "invalid rule" error.

  [#332831]

- When AAA-TM logs users off after their sessions time out, the traffic management session associated with the user is not terminated. If the number of abandoned traffic management sessions exceeds internal limits, the NetScaler ADC might become unresponsive.

  [#481876]

- In NetScaler 9.3 and previous versions, the NetScaler ADC used a SNIP address as the source IP address for authentication requests unless the administrator configured a static route to a different interface. In NetScaler 10.1 and subsequent versions, the ADC uses the NSIP address as the source for authentication requests even when a static route points to a different interface.

  To force the ADC to use a SNIP (not the NSIP) as the source IP address in version 10.1 or later, you can set up a load balancing virtual server with an authentication service, and then configure that load balancing virtual server to perform the authentication.

  [#457817]

- The AppFlow exporter might not export the correct information. Therefore, the client IP address shown on the NetScaler Insight Center dashboard might be incorrect.

  [#396892]

- AppFlow records generated by the NetScaler appliance cannot be seen on SPLUNK.

  [#327439]

- The following behavior is seen during a high availability failover on a NetScaler appliance that has active ICA session applications launched:

  --- The applications stop functioning, but are visible on the browser.

  --- The Citrix Receiver displays a dialog box, with a message stating that the connection is disconnected.

  --- When you click OK on the dialog box, the applications are not displayed anymore.

  --- If you launch any fresh applications without re-login, all the previously launched applications will resume with the previous status.

  [#388563, 438710, 488206]

- The timestamp in AppFlow records are not in NTP format.

  [#525568]

- If a user request triggers an application firewall policy that is bound to the APPFW_BYPASS profile, the application firewall might fail to generate an SNMP alarm.

  [#489691]

- If the user sends a request that contains the string "Javascript" without a non-alphanumeric delimeter, the Cross-Site Scripting check does not block the request. This is expected behavior. Without a delimiter, the keyword "Javascript" cannot trigger code execution and therefore poses no threat to the protected web application.

  [#457926, 506333]

- The auto-update operation restores the default SQL/XSS patterns in the signatures. If the user edits a signature to remove any of the SQL/XSS patterns, the removed patterns might reappear in the signature when it is auto-updated.

  [#455652]

- On a NetScaler ADC that has the application firewall enabled and the HTML SQL injection feature configured to block, when the ADC detects an SQL violation on a page with a web form, a second violation might be generated for the Form Action URL. This is expected behavior. To avoid unexpected blocks, when you configure a relaxation for a web form, be sure to include a relaxation for the Form Action URL as well.

  [#451014]

- A NetScaler ADC that has the application firewall feature enabled might reset the connection after a protected web server issues an HTTP 204 response.

  [#427798]

- When you enable the sessionless URL closure feature, you must also enable the URL closure feature. If you do not enable URL closure, the sessionless URL closure feature does not work.

  [#283780]

- If the server sends less data than the amount specified in the Content-length header, the NetScaler application firewall might send a 9845 response and reset the connection.

  [#506653]

- On a NetScaler ADC that has the application firewall enabled and the buffer overflow check configured to block, the following error message might appear in the logs: "Internal error: additional data generated after partial response <blocked>." This error message indicates that a partial response was sent before the remainder of the response was blocked.

  [#498912]

- The Perl script that parses and merges the application firewall signatures during schema version upgrade can cause Perl to crash on the NetScaler ADC. These crash files can fill up the space on the hard drive, preventing access to the Graphical User Interface.

  [#532248]

- A POST request with an attached word document is silently blocked by the application firewall for a customized application.

  [#530277]

- URL Transformation, SSL VPN, and CVPN features leverage the application firewall processing engine and enforce the content-length check of the built-in dummy application firewall profile. For some transactions, this check truncates the processed data.

  [#532338, 526029, 539487]

- On a NetScaler appliance that has standalone application firewall license, when you bind a classic application firewall policy to a load balancing virtual server, an error message is displayed in the graphical user interface. The binding operation is successful. The error message is harmless and can be safely ignored.

  [#522712]

- For some malformed requests, the NetScaler application firewall log messages might not include the client IP address.

  [#510006]

- In NetScaler 9.3, if there is a standalone application firewall license, the user is able to bind a classic application firewall policy to the load balancing virtual server. However, in NetScaler 10.1, the design is changed. If the load balancing feature is not licensed, binding a classic application firewall policy to the load balancing virtual server now results in an error message.

[#510509]

- The customer's application does not work when the application firewall is deployed to inspect the request for security check violations. When the application firewall forwards the request to the backend server, the server responds with a 403 HTTP error code, indicating that it cannot properly validate the CORBA session, and sends the page without the expected data in the form fields. The root cause is under investigation.

  Workaround: Turn off form field tagging and credit card checks.

  [#511254]

- During an upgrade of a NetScaler appliance from version 10.0 to version 10.1 (build 121.1 or subsequent), the default JSON content type is not automatically configured. The default JSON content type is configured when version 10.1 (build 121.1) is installed on new hardware or in a new VPX instance. To check whether your appliance or instance has the correct default setting, log onto the NetScaler command line and type the following command:

  show appfw JSONContentType

  If the default content type is configured, the command output is similar to the following example:

  > show appfw JSONContentType

  1) JSONContenttypevalue: "^application/json$" IsRegex: REGEX

  Done

  If it is not, the screen shows only the following:

  > show appfw JSONContentType

  Done

  To add the default content type to the configuration, after upgrading to 10.1 (121.1), log onto the NetScaler command line, and then type the following commands to configure the default content type and verify the configuration:

  add appfw JSONContentType ^application/json$ -isRegex REGEX

  show appfw JSONContentType

  [#430014]

- If the application firewall blocks a request because of a limiting policy, such as a maximum upload size limit on a web form, the blocking action is not logged. If a custom redirect page has been configured for that web page, the application firewall does not display it.

  [#466329]

- With TCP services reachable over a GRE tunnel (without IPSEC), one or both tunnel end points (NetScaler appliances) might become unresponsive while monitoring the services over the tunnel.

  [#508535]

- In a cluster setup, a command that is executed on the cluster configuration coordinator is propagated to the other cluster nodes. Therefore, a command that takes a long time to complete (such as "save ns config"), can take a little extra time to complete on all the cluster nodes. During this time, if you execute another command on the cluster (through another session), that command will fail because the previous command is not yet complete.

  [#551607, 495270, 562651]

- In a cluster setup, the "show ns trace" command displays the trace only of the cluster configuration coordinator node. It does not show the trace of the other cluster nodes.

  [#568518]

- During an upgrade from a NetScaler 10.1 build to a NetScaler 10.5 build, running the "show audit messages" command can cause the NetScaler appliance to fail.

  [#546038]

- The NetScaler command line interface exits abruptly upon executing the "show dns addRec -format old" command.

  [#512526, 527066, 545578]

- When you use the Net::SSH::Perl library, and execute a command where an argument has a @ character, the NetScaler gives an error message indicating that the argument does not exist.

  For example, if you use the @ character in the tacacsSecret parameter of the following command:

  > set authentication tacacsAction TACACS-0101 -tacacsSecret Sl4make5f0rd@enc5

  Workaround: Use one of the following alternate approaches to execute the command:

  - Use Net::SSH::Perl library and include double quotes around the command when calling $ssh->cmd().

  - Use the Net::Telnet library.

  - Use the Net::SSH::Expect library.

  [#346066]

- The "alias" command prepends an extraneous quote character. As a result, the command does not work as expected.

  [#531114]

- If you open the NetScaler ADC configuration utility on multiple browser tabs, and if you disable a feature on one of the tabs, the other tabs are not automatically refreshed.

  Workaround: Manually refresh the tabs.

  [#469755]

- The Upgrade Wizard sometimes does not display a message when the appliance is rebooting. However, the NetScaler appliance reboots and the upgrade is successful.

  Workaround: Reload the NetScaler GUI through the browser.

  [#557379, 585649]

- If you use a Chrome browser to access the NetScaler graphical user interface (GUI), the browser might display the Page Unresponsive error message.

  Workaround:

  If you are using a Windows computer, do the following:

  1. Right-click the shortcut icon that you use to open the Chrome browser, and select Properties from the pop-up menu.

  2. In the Google Chrome Properties dialog box, click the Shortcut tab and, in the Target field, append the following value: --disable-hang-monitor

  For example: "C:\\Program Files (x86)\\Google\\Chrome\\Application\\chrome.exe --disable-hang-monitor" http://www.google.com

  3. Close all instances of the Chrome browser, and restart the Chrome browser.

  If you are using a MAC computer, do the following:

  1. Open the terminal.

  2. Launch the Chrome browser from the terminal and append the --disable-hang-monitor value, as follows:

  open -a /Applications/Google\\ Chrome.app --args --disable-hang-monitor

  [#400073, 401262]

- If, when using the configuration utility to configure a NetScaler ADC, you press Alt+Tab to switch between programs, the current dialog box might disappear, hidden behind the main configuration utility screen. To reach the dialog box, press Alt+Tab a second time.

  [#374437]

- On the Reporting tab of the NetScaler GUI, if you have chosen to use the time zone settings of the NetScaler ADC, the System Overview graph does not reflect the time zone set on the NetScaler ADC. The values in the graph are for the GMT time zone.

  [#485314]

- The Traffic Management > Load Balancing > Set up NetScaler for XenApp/XenDesktop wizard displays an error if more than one service group is bound to the virtual server that is used for load balancing the XenApp/XenDesktop servers, or if more than one service is bound to the service group.

  [#414807]

- You cannot use the configuration utility to add signatures to an existing application firewall profile using the wizard, if the application firewall policy is not globally bound.

Workaround: Use the command line interface .

[#470941]

- The maximum length for creating a NetScaler ADC system user password (System > User Administration > Users) is 127. The GUI tooltip displays this value as 255, which is incorrect.

  [#499223, 561315]

- If you use the Google Chrome browser to access the NetScaler configuration utility, and the monitor resolution is low, you might not be able to use the mouse to scroll the screen.

  Workaround: Use the arrow keys on the keyboard to scroll the screen.

  [#389328]

- If you access the NetScaler configuration utility from the Start screen on a Windows 8 machine, the Java based configuration views are not displayed.

  Workaround: Switch to the Desktop screen to display Java based configuration views. Microsoft Windows 8 does not support plug-ins on the Start screen, and therefore Java cannot run on the Start screen. For more information, see http://www.java.com/en/download/faq/win8_faq.xml

  [#388534]

- In a high availability setup, if you run the "add ssl certkey" command on the primary node, and the certificate and key files are not present on the secondary node, the command fails?on the secondary node. However, the configuration utility does not display an error message.

  [#459703]

- When using the Traffic Management > Load Balancing > Set Up NetScaler for XenApp/XenDesktop wizard, Web Interface on NetScaler does not publish XenDesktop applications if the load balancing virtual server is configured to listen on two XenDesktop servers.

  [#414422]

- In certain cases, if the state of a load balancing virtual server changes, the NetScaler appliance might fail while changing the state of the associated content switching virtual server.

  [#522510, 528782, 538223, 552913]

- When you configure load balancing virtual servers in a content switched environment, the service types of primary and backup virtual servers must be the same. If you assign a backup virtual server with a service type of TCP to a load balancing virtual server with a service type of HTTP, any content switching action bound to the load balancing virtual server fails.

  [#399575]

- If DNS caching is enabled and the NetScaler ADC receives a query that is not cached, it forwards the query to the name server. It sends the response from the server to the client and also caches the records in the Answer, Authority, and Additional sections of the DNS response. The response from the server can have the AA bit set or unset.

  - If the AA bit is set and a query is received for a record that was cached and a part of the Authority or Additional section, the ADC responds to the query from its cache, with the AA bit unset and TTL decremented.

  - If a subsequent query is received for a record that is cached and was part of the Answer section, the ADC responds to the query from its cache, with the AA bit set and the original TTL.

  [#458244]

- If, while adding a DNS record (such as addrec and nsrec) from the GUI or by using the NITRO API, you specify the TTL value as 3600, the value of the minimum TTL of the SOA record is used instead.

  [#382478]

- A NetScaler client becomes unresponsive if:

  1. The NetScaler appliance receives the complete response to the client's query from the server.

  2. At the same time, the client sends an attention packet to the appliance.

  The client becomes unresponsive because the appliance closes the server-side connection but does not send the client a response to the attention packet.

  [#560401]

- In all releases of 10.0 and 10.1, the "show server" output does not include IP address and state information for GSLB services.

  This feature works in all builds of the 9.3 and 10.5 releases.

  [#499523]

- GSLB force sync option fails, if the following conditions are met:

  * The same load balancing (LB) monitor is bound to a GSLB service as well as other LB entities.

  * The server IP address already exists in the slave node under non-GSLB entity (the entity with same server IP address but with different server name) and the master node tries to synchronize the configuration.

  [#530638, 506432]

- If you do not specify the deployment details when you import the SharePoint AppExpert template, you cannot configure backend servers.

  [#511638]

- When upgrading HA nodes that have Web Interface on NetScaler (WIonNS) build 126.x, the updates made in the Webinterface.conf file are overwritten by the previous version of the file. This is due to the rolling upgrade of HA nodes or due to the file sync operation between HA nodes.

  To avoid this issue, use the following steps when upgrading the HA nodes:

  1. Before upgrading, run the "set ns param -internaluserlogin DISABLED" command.

  2. Upgrade the secondary HA node to NetScaler release 10.1 build 126.x.

  3. Force failover to make the upgraded node the primary node.

  4. Upgrade the other HA node to NetScaler release 10.1 build 126.x.

  5. Reenable the "internaluserlogin" parameter with the "set ns param -internaluserlogin ENABLED" command.

  6. Save the configurations.

  Note: Before upgrading synchronize files between the HA nodes by using the "sync ha files all" command.

  [#471294]

- In a NetScaler deployment that has integrated caching and SSL enabled, the NetScaler can crash in the following scenario:

  1. Client1 requests for an object that is not in cache.

  2. While the NetScaler fetches the object from the backend server, client2 (a slow client) sends a request for the same object.

  3. Client1 now decides to reset the connection.

  4. When available, NetScaler serves the object to the client2.

  However, since client2 is slow, large data is piled up on the NetScaler that needs to be forwarded to client2. When the NetScaler tries to send this large data to the client, the NetScaler can crash.

  [#486535]

- When a selector-based content group has been configured, the NetScaler ADC can fail when a policy associated with this content group is matched and the response status is "404 Not Found".

  [#440107, 440389]

- If a NetScaler appliance sending a DNSSEC negative response over UDP is not able to include the required records (for example, SOA, NSECs, and RRSIG records) in the Authority section, the appliance might send a truncated response in the wrong packet format.

  [#540965]

- The NetScaler ADC might fail or become unresponsive if the FTP virtual server name exceeds 32 characters and L2Conn is enabled on the virtual server.

  [#441776]

- If a DNS autoscale service group is bound to a virtual server, the "show lb vserver" command output displays one extra service bound to the virtual server.

  [#464952]

- When you load balance secure XenApp servers running SSL,using Netscaler load balancing servers, then Constrained Delegation between WebInterface and XenApp fails.

  [#535103]

- If the FQDN is not resolvable, you might notice high CPU utilization on the NetScaler ADC.

  [#455133]

- The appliance fails if non-reachable autoscale entities that are part of a service group later become reachable and, in the interim, the service group name has changed.

  [#583647]

- The session timeout value that is set for a particular user (using the "add/set system user -timeout <value>" command) is not used as the session timeout value for that user when logged on to a NITRO session in NetScaler 10.1 builds. The session timeout value for NITRO API can be specified at login time, otherwise a default value of 30 minutes is taken as the session timeout value.

  Note: In NetScaler 10.5 builds, the user session timeout value is by default used as the NITRO API session timeout value for that user.

  [#513938, 525094]

- Configuring singleton entities such as lbparam, sslparam, csparam, and vpathparam by using the "application/vnd.com.citrix.netscaler.<entityname>+json" content-type, results in error. For example, you get an error when setting the vPath parameter as follows:

  - URL: /nitro/v1/config/vpathparam

  - Method: PUT

  - Content-type: application/vnd.com.citrix.netscaler.vpathparam+json

  - Request payload: {"vpathparam":{"encapsulation":"enabled"}}

  - Response: {"errorcode": -1,"message": "Entityname is missing","severity": "ERROR"}

  Workaround: Use one of the following content types to resolve the issue:

  Option1: Using the "application/json" content type.

- URL: /nitro/v1/config/vpathparam

- Method: PUT

- Content-Type: application/json

- Cache-Control: no-cache

- Request payload: {"vpathparam":{"encapsulation":"enabled"}}

Option2: Using the URLEncoded content type.

- URL: /nitro/v1/config/

- Method: PUT

- Content-Type: application/x-www-form-urlencoded

- Cache-Control: no-cache

- Request payload: {"vpathparam":{"encapsulation":"enabled"}}

[#574321]

- The "set appfw" command cannot be executed on the Netscaler ADC if TACACS server is used for authorization. An error message -"Not authorized to execute this command" might be seen.

  [#519898]

- In a Double-Hop deployment, if a STA server on the first hop is DOWN, the ns.log file is filled with SSLVPN Message: "Sent IPv4 Socks connect reply to client. Connection Refused "

  [#559879]

- EULA should not be prompted when interface type is modified from Shared to Passthrough for a NetScaler-VSB provisioned on Nexus 1010/1110 platforms.

  [#471373]

- Log file gets congested with the messages reporting WI monitor services as UP, with no corresponding DOWN messages.

  [#562819]

- When running the NetScaler Gateway wizard, a prompt appears at the end to make the LDAP authentication policy the primary authentication type, even though the LDAP policy is selected as primary earlier in the wizard.

  [#427092]

- The pop-up messages for NetScaler Gateway Plug-in for Windows appear behind the active applications (such as browsers) on Windows 8.

  [#511757]

- A customer reported: with TACACS+?configured for AAA, and authorization disabled on the policy when the customer runs commands, they get a "Not Authorized" error.

  [#573836]

- The NetScaler counters, used to verify connected users, display a value that does not reflect actual connections.

  [#490991, 398874]

- During login, the client authenticates, but the kernel module did not compile.

  Cause and solution: The Linux NSGClient expects linux kernel headers to be installed. The user needs to install linux headers manually before installing NSGClient. The command would be,

  apt-get install linux-headers-`uname -r`

  [#545810]

- For windows, if the NetScaler VPN plugin is auto-upgraded from version 9.3 or older, and the Citrix receiver is running, after the auto-upgrade the home pages are displayed in two tabs.

  [#530310]

- If you apply the Citrix Receiver theme to the NetScaler Gateway logon page, the layout appears garbled on computers running Windows XP Service Pack 3 with Internet Explorer 7 browsers.

  [#346729]

- On an nCore appliances, when users attempt to access the subnet IP address through the VPN tunnel over HTTP, a 401 Access Denied error message appears. Connecting to the subnet IP address works if users make the attempt by using HTTPS.

  [#373991]

- If you configure NetScaler Gateway as a high availability pair and if there is a failover from the primary to the secondary appliance, the ICA connection to published apps that are already open on the user device is reestablished. If users attempt to open more applications from the Web Interface, the applications fail to open and user receive an error message

  [#384998]

- When users log on with the NetScaler Gateway Plug-in, when WiFi roaming occurs, intermittent ICMP requests time out and users cannot access network resources.

  [#392389]

- If Pre-authentication scan is configured on NetScaler and users launch NetScaler Gateway plugin when browser is already opened then users intermittently get redirected to "Internal error" page.

[#393357]

- If you configure an intranet IP address, when users log on by using clientless access and then open SharePoint 2007, when they try to open a folder with Windows Explorer, a blank page appears.

  [#376303, 394800]

- Installing and uninstalling the NetScaler Gateway Plug-in can take a long time. This is due to multiple entries of the Citrix Virtual Adapter in the registry.

  [#398693]

- Java Plug-in: Intranet Application fails to connect if AG VIP is running on non default port

  [#399405]

- The NetScaler Gateway Plug-in for Java does not compress network traffic even when compression is configured on the NetScaler Gateway.

  [#400050]

- If you configure Group Extraction, you cannot bind an LDAP authentication policy to the virtual server. However, if an LDAP authentication policy was bound to the virtual server before configuring Group Extraction, you can enable authentication on the virtual server.

  [#400171]

- If you configure a NetScaler Gateway virtual server, enable ICA proxy and enable the Use Source IP (USIP) mode globally, when users connect and use StoreFront to open an application, NetScaler Gateway uses the client IP address as the source IP address when contacting the STA server and the application fails to open. If you disable USIP mode, the same behavior occurs unless you restart the NetScaler Gateway appliance. To avoid the issue, you need to configure a service on NetScaler for the STA server and disable USIP on that service.

  [#411851]

- When you use the Set Up NetScaler for XenApp/XenDesktop wizard in NetScaler, apply optimization settings, and bind the cache policy globally, when users log on with the NetScaler Gateway Plug-in and open Citrix Receiver, the applications and desktops do not appear. The following message appears: There are no apps or desktops assigned to you at this time. Citrix recommends disabling the optimization settings.

  [#411152]

- If you configure the appliance with NetScaler Gateway and Application Firewall, logon attempts by unauthorized users appear in the logs. When an authorized user logs on and then attempts to access a network resource to which users are explicitly denied, the access attempt does not appear in the logs and users receive a 403 error.

  [#374890]

- When users connect, the DNS Service Location (SRV) records configured on NetScaler Gateway are not served.

  [#464518, 467420]

- To configure two-factor authentication with SAML authentication, you must configure the secondary authentication

policy in the primary cascade.

[#397625]

- Occasionally, the NetScaler shows, "Error: Not a privileged user", when attempting to reach ShareFile through the SAML method.

  [#555779]

- A NetScaler Gateway NSPPE crash happened under the following conditions:

  * wihome is configured with FQDN

  * wihome is removed immediately even before the DNS response for the same comes back

  [#542560, 566864]

- If you have configured a proxy server and you configure NetScaler Gateway to route traffic through the proxy server, when users log off from a clientless access session, a 403 error occurs.

  [#385318]

- The running configuration does not include group extraction policies bound to the NetScaler Gateway virtual server.

  [#368229]

- The URL rewrite label, which is set at the global level, overrides the rewrite label that is set at the virtual server level. The settings in the global level should not override the virtual server level settings.

  [#444715]

- In transparent mode, after you initiate a session and launch an application through Citrix Receiver (Enterprise edition) from a Windows 8 client, the session terminates and resumes when you launch subsequent applications. Consequently, HDX Insight reports include session termination records.

  [#399626]

- On the Dashboard > HDX Insight > Users page, the report for user sessions displays incorrect values. The left pane displays the average values for the entire session, but the right pane displays the values for the period selected from the drop-down list.

  [#397236]

- The time values on the graphs display overlapping values, mostly in the 5-minute-interval view.

  [#379876, 424686, 437964]

- In a graph that displays a very low number of data points, the time value displayed on the x-axis includes milliseconds. The value displayed for milliseconds has no significance.

  [#368967]

- On the Dashboard > Web Insight > Applications page, the values shown when you select "Response Time" from the

drop-down list can be incorrect.

[#394526]

- If an ICA session is initiated by launching XenDesktop, the user name is displayed along with the domain name "(user-id@domain-name)".

  [#385821]

- The following error message appears when NetScaler Insight Center installed on VMware ESX is powered on or off:

  The VMware Tools power-on script did not run successfully in this virtual machine. If you have configured a custom power-on script in this virtual machine, make sure that it contains no errors. You can also submit a support request to report this issue.

  [#414160]

- In some instances, the bar line on a graph appears outside the time points on the x-axis.

  [#446120]

- On the HDX Insight reports, a Y-axis value of 0 is sometimes shown at a location higher than the x axis.

  [#414214]

- When launching n instances of an application, the NetScaler appliance sends n-1 termination records for the application. Consequently, the HDX Insight node displays only a single instance of this application as active.

  [#386911]

- Upgrading NetScaler Insight Center on a VMware ESX server from build 118.7 or 119.7 to build 120.13 or later is not supported. However, upgrading from build 120.13 to later build is supported.

  Workaround: To upgrade to build 120.13 or later, perform a fresh installation. To retain your existing configurations, make sure that the IP address of the NetScaler appliance and the IP address of NetScaler Insight Center remain the same.

  [#424673]

- NetScaler Insight Center might not display reports under the following set of conditions:

  -NetScaler ADCs that are configured for Network Address Translation (NAT) are added to the NetScaler Insight Center inventory.

  -A NetScaler ADC and a NetScaler Insight Center virtual appliance are in different networks and are configured for NAT.

  [#441163]

- When you launch XenApp through Citrix Receiver (standard edition), the app launch duration is not calculated and is shown as zero.

  [#388096, 423109]

- In SDX systems, sometimes interface or channel binding to a VLAN fails. This happens only if the interface is down or one of the member interfaces of a channel is down.

  [#474438, 493664]

- On adding many VPX instances, you may hit the default cache memory limit which could result in unexpected behavior.

  Workaround: Increase the default cache memory limit.

  [#499311]

- Configuring of Enforced VLAN by providing VLAN number in the VLAN ID field is not supported on SDX Rome 1G port.

  Workaround: Use the Allowed VLAN feature which has the same functionality as Enforced VLAN.

  [#369650, 442942, 468381]

- On SDX systems, sometime while creating/deleting or modifying a LACP channel, TX stalls are seen on some of the member interfaces.

  [#476304]

- If there are no 0/x interfaces provisioned in a VPX and 10/x or 1/x interface is used for manageability purpose and the user creates a LACP channel on those interfaces (10/x or 1/x), then the VPX will become unreachable by Management Service and the channel configurations will not be propagated to the VPX.

  The best way to achieve the same is to create the required channel first and then provision the VPX with the LACP selected.

  [#506167]

- If you use the Management Service to bind a new interface to an LACP channel or unbind an existing interface, all the member interfaces of the LACP channel are reset. This forces an HA failover.

  [#434687]

- If you disable an interface of an LA channel configured on a NetScaler instance running on a NetScaler SDX appliance, the SDX appliance does not notify the peer device that the interface is disabled. Therefore, the peer device might send traffic to the disabled interface.

  Workaround: Disable the interface of the peer device so that it does not send traffic to the disabled interface of the SDX appliance.

  [#384909]

- On a NetScaler VPX instance running on a Linux-KVM platform, dynamic routing protocols OSPF and ISIS fail to run on the platforms MacVTap interfaces.

  Workaround: Enable promiscuous mode on these MacVTap interfaces, using either the Linux-KVM graphical interface (Virt-Manager) or the Linux-KVM command line interface (virsh).

  [#405164]

- A NetScaler VPX instance might fail to restart on a Linux-KVM virtualization platform using processors that do not support the constant_tsc CPU feature.

  [#405383, 360482]

- The NetScaler appliance does not create session entries for ICMPv6 packets that match a forwarding-session rule.

  [#399436]

- A large number of IPv6 client connections (more than 2 million) can degrade the performance of a NetScaler appliance.

  [#575126]

- In an active-active high availability configuration using Virtual Router Redundancy Protocol (VRRP) protocol, a ping to a virtual IP address (VIP) might fail from a node that is a backup node for this VIP address.

  [#485260]

- In a high availability configuration with the network firewall mode set to BASIC on the current secondary node, synchronization of configuration files from the primary to secondary node fails, regardless of whether you run the "sync HA files" command from the NetScaler command line or by using the Start HA files synchronization dialog box in the configuration utility.

  Workaround: Add the following extended ACL on each node of the HA configuration:

  > add acl <aclname> -srcIP <NSIP of the peer node> -protocol TCP -destport 22

  For example, for an HA configuration in which the primary node's NSIP address is 198.51.100.9 and the secondary node's NSIP address is 198.51.100.27, you would run the following commands:

  On the primary node:

  > add acl ACL-example -srcIP 198.51.100.27 -protocol TCP -destport 22

  On the secondary node:

  > add acl ACL-example -srcIP 198.51.100.9 -protocol TCP -destport 22

  [#371613]

- The NetScaler ADC might become unresponsive if you run the show route operation during a dynamic route addition or deletion process.

  [#323127]

- For an RNAT connection, the NetScaler appliance drops the first packet that the server sends to the client.

  [#543171]

- In an HA configuration in INC mode running the OSPF routing protocol, the secondary node drops all L3 traffic that has the destination that was advertised by the secondary node.

  [#318684]

- Live migration of a NetScaler virtual machine running on a Linux-KVM host is not supported.

  [#407185]

- A high number of nic_err_rx_crc errors has been attributed to improperly seated or faulty small form-factor pluggable (SFP) transceivers.

  Workaround: If experiencing nic_err_rx_crc errors, perform a manual diagnostic check to rule out problems with SFPs, cables, and connectivity with the partner device ports.

  [#494183]

- In rare conditions, a 10G interface might stop processing the traffic.

  Workaround: Reset the interface.

  [#519000, 519041]

- On the MPX 8200/8400/8600 and MPX 5550/5650 platforms, if a 1G data port is connected but disabled, the status of the peer port on the switch might be shown as UP after the MPX appliance restarts.

  [#385217]

- If you add an NTP time server by specifying the server name (host name), and the ns.conf file is very large, the result is a race condition in which the NTP daemon (NTPD) is started before host name services are ready.

  Workaround: Do one of the following:

  -Restart the NTP daemon after starting the NetScaler appliance.

  -Add the NTP server by specifying the IP address of the server instead of specifying the host name.

  [#573306]

- In rare conditions, Network Interface Cards (NIC) might become unresponsive and reset polling on these interfaces also might fail.

  [#535789]

- L2 mode is not supported on NetScaler VPX instances running on a Linux-KVM host.

  [#402113]

- VLAN tagging is not supported on a Netscaler VPX instance operating in MacVTap-Bridge, MacVTap-Private, MacVTap-VEPA, or MacVTap-Passthrough interface mode.

  [#402111]

- The memory usage statistic shown on the LCD display of a NetScaler appliance is the allocated memory. The NetScaler configuration utility displays the currently used memory. Therefore, the two values are different.

  [#334358, 576545]

- LACP is not supported on Netscaler VPX instances operating in Bridge, MacVTap-Bridge, MacVTap-Private, or MacVTap-VEPA interface mode.

  [#407184]

- You cannot use the configuration utility to define classic SSL policies. However, you can use the configuration utility to bind and unbind classic SSL policies.

  Workaround: Use the CLI to define classic SSL policies.

  Note: Citrix encourages the use of default syntax policies rather than classic policies.

  [#390584]

- Some IP based expressions might not work for IP addresses starting from octet 128 or greater (128.x.x.x - 254.x.x.x).

  The following expressions are not impacted:

  - EQ

  - IN_SUBNET

  - IS_IPV6

  - GET1

  - GET2

  - GET3

  - GET4

  - MATCHES

  - MATCHES_LOCATION

  - APPEND

  - TYPECAST_TEXT_T

  - TYPECAST_IPv6_ADDRESS_AT

  The following expressions do not work:

  - GT

  - GE

  - LT

  - LE

  - BETWEEN

- NE

- ADD

- SUB

- MUL

- DIV

- MOD

- NEG

- BITAND

- BITOR

- BITXOR

- BITNEG

- LSHIFT

- RSHIFT

- TYPECAST_TIME_AT

- TYPECAST_IP_ADDRESS_AT

- TYPECAST_DOUBLE_AT

- TYPECAST_UNSIGNED_LONG_AT

- WEEKDAY_STRING

- WEEKDAY_STRING_SHORT

- SIGNED8_STRING

- UNSIGNED8_STRING

- SIGNED16_STRING

- UNSIGNED16_STRING

- SIGNED32_STRING

[#534244]

- If a wildcard virtual server (** IP address and port values) that accepts both IPv4 and IPv6 packets uses a listen policy of CLIENT.IP.PROTOCOL.EQ(ICMP) to capture ICMP traffic, it also captures IPv6 packets in which the second byte of the source IPv6 address has a value of 01).

  Workaround: First use an expression that filters the IPv4 traffic, and then use an expression that reads the protocol value from the filtered IPv4 packets and checks for a protocol value of ICMP.

!CLIENT.IP.SRC.IS_IPV6 && CLIENT.IP.PROTOCOL.EQ(ICMP)

[#422967]

- After you import a custom data source, the charts for the counters under "System entities statistics" are inaccurate, because of issues in the third party charting engine.

  [#368982]

- If you disable SSLv3 on the "nskrpcs-127.0.0.1-3009" service, an "ERROR: Operation not permitted" message appears even though SSLv3 has been successfully disabled on the service.

  [#521569]

- If you update the certificate-key pair for a service group, the change is not reflected in the individual services that are bound to this service group. As a result, the old certificate-key pair continues to be used for negotiation in the SSL handshake.

  [#554925]

- An incoming SSL record that spans more than 256 TCP packets and contains TCP header options causes memory corruption in the Cavium command buffer structure. As a result, the NetScaler appliance fails.

  [#573904, 583295, 590222]

- In rare cases, the "update ssl certKey" command fails and, in spite of displaying a "Resource already exists" error message, creates a stale duplicate entry with the same certificate-key pair in the configuration file (ns.conf).

  [#519368]

- If a certificate has a validity of 100 years, Days to Expiration incorrectly appears as 0 in the NetScaler command line interface and the configuration utility.

  [#509608]

- If the format of a CRL is incorrect or the issuer of a CRL does not match the specified CA certificate, and you run the "show crl" command, an error message showing the CRL status as invalid appears.

  [#468198]

- In a cluster setup, if you include the "cipherdetails" option in the "show ssl service" or "show ssl vserver" command, an incorrect message appears. This is only a display issue.

  For example,

  > show ssl service svc1 -cipherDetails

  ERROR: No such resource [serviceName, svc1]

  [#402423]

- On a NetScaler MPX or SDX appliance, TLS protocol version 1.2 does not support a client certificate with an RSA 4096-bit key.

  [#343395]

- An SSL chip is disabled at the third reinitialization attempt. That is, the maximum reinitialization limit is 2. Earlier, this limit was 5.

  [#455821]

- When the management CPU is running at close to 100% of capacity, the aggregator might not be able to process some of the statistics requests from clients, such as requests from the configuration utility, the CLI, and SNMP. If the aggregator fails to respond within the timeout period, the client returns following error:

  Invalid response from the aggregator [Device not Configured]

  [#377618, 341460, 351127, 364015, 481575, 499259]

- FTP connections through a TCP wildcard virtual server on the NetScaler appliance might fail for one of the following reasons:

  - A mismatch in TCP parameters is preventing the appliance from reusing the probe connection.

  - The server is sending data before the client-side TCP connection is established.

  [#545858]

- When SPDY Protocol is enabled and SPDY Traffic is received on the NetScaler appliance, the TCP current clients counter goes to negative values and shows a very large value in the stat or the SNMP OID.

  [#551562, 551786, 568554]

- If a client on an IPV6 connection advertises an MSS value below 1360 (bytes), the NetScaler appliance responds with a MSS value below the (RFC) required minimum value of 1220 (bytes).

  [#556475]

- The NetScaler appliance displays cluster related logs even if it is not in a clustered configuration or does not have a cluster license.

  [#543429]

- The initial client connection on the NetScaler appliance might fail if a wildcard virtual server is configured and the useProxyPort option is disabled globally on the appliance.

  [#542776, 571357]

- You cannot generate a report from the imported log files.

  [#535527]

- In a cluster or HA setup, when you perform an operation that adds a new file (create/import SSL/APPFW), the files is

synchronized to the other nodes (non-CCO nodes in a cluster or the secondary appliance in an HA setup). This synchronization either happens either periodically or when manually executed. If an operation that uses this file is executed before the file is synchronized, the operation fails, because the required file is not available.

For example, if you import a certificate file, and then execute the "show cert key" command immediately, the command fails.

This issue is fixed by synchronizing the files across all the nodes automatically, after they are added.

[#535162, 288743, 389394, 470729, 562724]

- Load Balancing

  If all of the following conditions are present, they might lead to a situation in which CPU usage is significantly different among the packet engines (PEs):

  1. The maximum number of clients (maxclients) for a service is set to a value less than the number of PEs in the system.

  2. Connections to this service have a high degree of connection reuse, that is, multiple requests are sent on the same TCP connection.

  3. Requests for connections to this service cause a surge queue buildup.

  If the maxclient setting is less than the number of PEs, only some PEs can open connections. After the maxclient limit is reached, PEs that have open connections are not likely to close them, because they are using those connections to process the traffic generated by high connection reuse and the large surge queue. As a result, the other PEs might not be able to open new connections. They therefore have a lower level of CPU usage, because they cannot participate in processing the surge queue.

  This is expected behavior and usually does not cause any issues. However, if some of the PEs have near 100% CPU usage while the other PEs have relatively low CPU usage, you might want to limit the maximum requests per connection by using the "set service <name> -maxReq <positive_integer>" command, so that the PEs close connections that have delivered the specified number of requests. This evens out the CPU usage, because it allows the other PEs to open connections to the service.

  [#516606, 528242]

- The virtual IP (VIP) address of a load balancing virtual server cannot be changed if the LB virtual server and syslog server have same configuration (ip, port, service) and use the same server information. In such cases, if the syslog server's IP address is changed, the syslog server uses different server information and does not update the server information used by the LB virtual server. As a result, the LB virtual server displays an error message when you try to change its VIP address.

  [#522665]

- In deployments with large configurations (in the order of 2 MB), when the load on the management CPU is high, the execution of the "show ns runningConfig" command can take a large amount of time.

  Workaround: If you're executing the command manually, then there is no workaround. However, if you are using a script to fetch the the output of the "show ns runningConfig" command, and if the script has a timeout, then modify the script to increase timeout to 500 seconds. The command could be executed within that time period.

  [#449234, 457629]

- When using MPTCP, if a single SSL record is split into a large number (> 100) of small segments, an SSL buffer overrun causes the NetScaler appliance to crash.

  [#427126, 441982, 452885, 456645]

- HA SYNC takes longer than expected for NetScaler 1000V. For example, for synchronizing ns.conf file of 38.4 KB size, it takes 70-100 seconds.

  [#508410]

- Setting 'Request timeout' or 'Request timeout action' in HTTP Profiles can cause the NetScaler to fail in some situations.

  [#501100]

- Every Domain Based Service (DBS) on a NetScaler appliance is assigned two monitors. Therefore, the limit of 7500 monitors can result in a memory allocation failure when you add a new service to the appliance.

  [#523473]

- On a NetScaler 1000V instance, transmit congestion occurs on virtual interfaces in high traffic conditions.

  [#430154]

- If an LACP channel is bound to nine or more interfaces and is a member of a tagged VLAN, deleting the channel from a service VM can cause the NetScaler appliance to fail intermittently.

  [#524320]

- Configuration Utility

  If, while upgrading a NetScaler appliance, you change the RSS key type, the configuration utility does not display a warning message to restart the NetScaler appliance.

  [#542702]

- Configuration Utility

  A large configuration file puts a heavy load on the management CPU. The resulting delay in displaying the output of the "show ns runningconfig" command might exceed the timeout value.

  Workaround: If you are using a script to fetch the output for "show ns runningConfig" command, and the script has a placeholder for timeout value, modify the script to increase the timeout value to 500 seconds.

  [#475830, 449234]

- On a NetScaler ADC, if WIHome is configured to point to an IPv6 load balancing virtual server that points to the IPv6 StoreFront services, a user trying to log on receives a 500 Internal Server Error message.

  Workaround: Remove the IPv6 load balancing virtual server configuration and configure WIHome to point directly to

the StoreFront server URL.

[#397150]

- The following APIs are not available in version 10.1 or later:

  - bindservicegroup_state2

  - unsetnslimitidentifier_selectorname. Use unsetnslimitidentifier_selector instead.

  [#363145]

What's New in Previous 10.1 Builds

Oct 10, 2015

The enhancements and changes that were available in NetScaler 10.1 releases prior to Build 133.9. The build number provided below the issue description indicates the build in which this enhancement or change was provided.

AAA Application Traffic | AAA-TM | AppExpert | AppFlow | AppQoE | Application Firewall | Cache Redirection | Cloud Integration | Cluster | Configuration Utility | Content Switching | DNS | DNS64 | DataStream | Global Server Load Balancing | Load Balancing | Load Balancing and AAA-TM | Monitors | NITRO API | NetScaler Gateway | NetScaler Insight Center | NetScaler SDX Appliance | NetScaler VPX Appliance | Networking | Platform | Policies | SNMP | SSL | Statistics | Support for ECDHE Ciphers | System

---

- Smart Group Option for LDAP Authentication

  When configuring AAA for LDAP, you can now set the default authentication group attribute explicitly, instead of allowing AAA to set the Group attribute from information that it extracts from credentials. In complex organizations that have multiple domains, smart group support allows simpler and more fool-proof implementation of SSO.

  To configure the smart group option at the command line, type the following command:

  > set aaa ldapParams -defaultAuthenticationGroup <string>

  For <string>, substitute the group identifier that you want to use.

  To configure the smart group option by using the configuration utility, in the Create Authentication Server or Modify Authentication Server dialog box, fill in the Default Authentication Group text box.

  [From Build 112.15] [#357837]

- Extracting Group Credentials from a Third Authentication Server

  When performing two-factor authentication, the AAA feature now supports extraction of the group membership credential from a third authentication server. This function is supported by use of a third authentication chain that is invoked only if the first and second authentication attempts succeed.

  To enable extraction of group membership credentials from a third authentication server, create an LDAP policy with authentication disabled. Then, bind that policy to the authentication virtual server, with the -groupExtraction flag set, as shown below.

  bind authentication vserver <name> [-policy <string> [-priority <positive_integer>]

  [-groupExtraction]]

  If -groupExtraction is set, the policy is an LDAP policy, and the policy has authentication disabled, then the policy is added to the third authentication chain. Otherwise, the binding will fail.

For more information, see http://support.citrix.com/proddocs/topic/ns-security-10-1-map/ns-aaa-setup-policies-authntcn-tsk.html.

[From Build 112.15] [#308118, 305567]

- Two-Factor SAML Authentication

  AAA now supports two-factor SAML authentication. When a user requests a resource, AAA checks for SAML policies. If a SAML policy with two-factor authentication is present, AAA redirects the user to the specified third-party authentication server. Once the user has authenticated and obtained a valid assertion, AAA redirects the user to the secondary login page for the resource.

  To enable two-factor SAML authentication, type the following command at the NetScaler command line:

  > add authentication samlAction <name> -samlTwoFactor ON

  For more information, see http://support.citrix.com/proddocs/topic/ns-security-10-1-map/ns-aaa-setup-policies-auth-saml-tsk.html.

  [From Build 112.15] [#277562]

- KCD Support for Microsoft SQL Data Stream

  Kerberos Constrained Delegation (KCD) is now supported for the Microsoft SQL server and the MSSQL data stream.

  For more information, see http://support.citrix.com/proddocs/topic/ns-security-10-1-map/ns-aaa-kerberos-kcd-con.html.

  [From Build 112.15] [#307491, 243724]

- Cluster Support

  Support for AAA-TM has been added to the NetScaler cluster when operated in spotted VIP mode. AAA-TM authenticates users correctly. AAA-TM commands run correctly at the command line, and the configuration utility displays the AAA-TM node and screens.

  [From Build 112.15] [#317306]

- Kerberos SSO

  The AAA-TM Kerberos functionality now supports single sign-on (SSO) with all supported authentication mechanisms. The CAC (Smart Card) and SAML SSO mechanisms are supported in all cases, regardless of the authentication method that the client uses to log onto the NetScaler appliance. The HTTP-Basic, HTTP-Digest, Forms-based, and NTLM (versions 1 and 2) SSO mechanisms are also supported if the client uses either HTTP-Basic or Forms-Based authentication to log onto the NetScaler appliance.

  You can configure Kerberos SSO to work in one of two ways: by impersonation or by delegation. To configure Kerberos SSO by impersonation, you must have the user's password or client certificate. To configure impersonation using a client certificate, the user must also have a properly-configured version of the Citrix Receiver installed on his or her personal computer. To configure Kerberos SSO by delegation, you must have the delegated user's credentials in one of the following formats: the user's password, the keytab configuration that includes an encrypted password, or the client cert and the matching CA certificate.

To configure Kerberos SSO, first configure your NetScaler appliance to manage traffic to the web application servers that users will access through SSO. Next, configure AAA-TM for your preferred authentication method. Verify that the NetScaler appliance can communicate with your LDAP Active Directory (AD) server and your Kerberos server.

What you do next depends on whether you want to configure Kerberos SSO by Impersonation or by Delegation. Follow the instructions in the appropriate section below.

Configuring Kerberos SSO by Impersonation

To configure Kerberos SSO by Impersonation, enable integrated authentication on each web application server. After you have done this, create and configure the NetScaler KCD account that will impersonate users.

To create the KCD account for SSO by impersonation with a password

At the NetScaler command prompt, type the following command:

add aaa kcdaccount <accountname> -realmStr <realm>

For each variable, substitute the following values:

* accountname - The KCD account name.

* realm - The domain assigned to Kerberos SSO.

Example:

add aaa kcdAccount kcdaccount1 ?realmStr EXAMPLE.COM

To create the KCD account for SSO by impersonation with a client certificate

At the NetScaler command prompt, type the following command:

add aaa kcdAccount <accountname> ?cacert <cacert>

For each variable, substitute the following values:

* accountname - The KCD account name.

* cacert - The full path and name of the CA certificate file on the NetScaler appliance.

Example:

add aaa kcdAccount kcdaccount1 ?cacert <path to certificate>

Configuring Kerberos SSO by Delegation

To configure Kerberos SSO by Delegation, next create an account (the Kerberos Service Account, or KSA) on the AD server for the NetScaler appliance to use as the delegated user. Next, in the KSA account Properties dialog box, Delegation tab, enable the following options: "Trust this user for delegation to specified services only" and "Use any Authentication protocol." Finally, add the HTTP service and any other services that Kerberos SSO will manage to the services list, which is located on the Properties tab beneath the two settings.

After you configure the NetScaler account on AD, enable integrated authentication on each web application server. Finally, create and configure the NetScaler KCD account that will serve as the delegated user.

To create the KCD account for SSO by delegation with a password

At the NetScaler command prompt, type the following commands:

add aaa kcdaccount <accountname> ?delegatedUser root -kcdPassword <password> - realmStr <realm>

For each variable, substitute the following values:

* accountname - The KCD account name.

* password - The password for the KCD account.

* realm - The domain assigned to Kerberos SSO.

Example (UPN format):

add aaa kcdaccount kcdaccount1 ?delegatedUser root -kcdPassword passsword1 -realmStr EXAMPLE.COM

Example (SPN format):

add aaa kcdAccount kcdaccount1 -realmStr EXAMPLE.COM -delegatedUser "host/kcdvserver.example.com" -kcdPassword password1

To create the KCD account for SSO by delegation with a keytab file

First, on the AD server, use the ktpass utility to create the appropriate keytab file. Next, use the file transfer utility of your choice to copy the keytab file from the AD server to the NetScaler appliance, and put it in /nsconfig/krb under the filename kcdvserver.keytab.

Next, at the NetScaler command prompt, type the following command:

add aaa kcdaccount <accountname> ?keytab <keytab>

Example:

add aaa kcdaccount kcdaccount1 ?keytab kcdvserver.keytab

Finally, verify that the new KCD account has the proper keytab file and virtual server principle associated with it:

To verify the KCD account on the NetScaler appliance

sh kcdAccount <accountname>

To create the KCD account for SSO by delegation with a client cert

At the NetScaler command prompt, type the following commands:

add aaa kcdaccount <accountname> -realmStr <realm> ?delegatedUser <spnuser> -usercert <cert> -cacert <cacert>

For each variable, substitute the following values:

* accountname - The KCD account name.

* realm - The domain assigned to Kerberos SSO.

* spnuser - The username in SPN format.

* usercert - The full path and name of the user client certificate file on the NetScaler appliance.

* cacert - The full path and name of the CA certificate file on the NetScaler appliance.

Example:

add aaa kcdaccount kcdaccount1 -realmStr EXAMPLE.COM -delegatedUser "host/kcdvserver.example.com" -usercert /certs/usercert -cacert /cacerts/cacert

[From Build 121.10] [#361257]

- Support for Additional Public Endpoints

  AppExpert applications and the deployment files created from them now support two or more endpoints. However, when importing an AppExpert template file, if you do not include a deployment file, the AppExpert Template Wizard displays a screen on which you can configure a maximum of two public endpoints: one endpoint of type HTTP and one endpoint of type HTTPS. So, if you want more than two endpoints, you have to configure additional endpoints after you create the application. You can then export the application to obtain a deployment file that contains all the configured endpoints.

  For information about importing an AppExpert template, configuring public endpoints after importing an application, and exporting an AppExpert to a template file, see http://support.citrix.com/proddocs/topic/ns-main-appexpert-10-1-map/ns-aapexpert-apptemp-get-started-tsk.html.

  [From Build 112.15] [#259600]

- Configure a Persistency Group for Application Units

  You can now configure a persistency group for the application units in an AppExpert application. In the context of an AppExpert application, a persistency group is a group of application units that you can treat as a single entity for the purpose of applying common persistence settings. When the application is exported to an application template file, the persistency group settings are included, and they are automatically applied to the application units when you import the AppExpert application.

  For more information about configuring a persistency group for the application units in an AppExpert application, see http://support.citrix.com/proddocs/topic/ns-main-appexpert-10-1-map/ns-appexpert-apptemp-config-pers-groups-for-app-units-tsk.html.

  [From Build 112.15] [#243716]

- Enhanced Target Support for RefineSearch Parameter in Rewrite

  The RefineSearch and Target parameters can now be used together in a single Rewrite action. The following types of search are supported:

  * TCP with regular expressions

  * HTTP with regular expressions

  * HTTP with XPath expressions

* HTTP body payload expressions

[From Build 112.15] [#245438, 243248, 247097]

- Service and Service Group Configurations Exported to Application Templates

  When you export an AppExpert application, all services and service groups that are part of the application configuration are exported to the deployment file. During import, the appliance compares the deployment file's contents with its own configuration, and manages conflicts in the following way:

  - If a service in the file has the same name and service type as a service on the appliance, the appliance does not import the service. It binds the existing service to all the application units created during import.

  - If a service in the file has the same name as a service on the appliance, but its service type is different, the appliance does not import the service. It displays a message indicating a protocol mismatch.

  - If a service in the file has the IP address and port combination of a service on the appliance, and both services use the same underlying transport protocol (for example, HTTP and SSL services both use TCP), the appliance does not import the service, even if their names are different. It displays a message indicating a port and service type conflict. If the IP address and port combination is same, but the name and underlying transport protocol are different, the appliance imports the service.

  - If a service group in the file has the same name and service type as a service group on the appliance, the appliance does not import the service group. It binds the existing service group to all application units created during import.

  - If a service group in the file has the same name as a service group on the appliance, but its service type is different, the appliance does not import the service group. It displays a message indicating a protocol mismatch.

  If a conflict is detected during import, the appliance ends the import process and rolls back any configuration changes that were made, preserving the configuration that was in place before the template was imported.

  [From Build 112.15] [#248273]

- AppExpert Template for Microsoft Outlook Web Access

  An AppExpert template has been created to help users configure the application firewall to protect a web server that runs Microsoft Outlook Web Access. The template and associated signatures file provide an appropriate default configuration for the application firewall when protecting OWA. The template is posted on the Citrix Community Web Site, and can be downloaded from within the configuration utility, in the main AppExpert pane, by clicking Download AppExpert Templates.

  To install the downloaded templates, first extract them from the archive to a temporary location on your local computer. The archive contains four files:

  * OWA_Template.xml?The actual template

  * OWA_signatures.xml?The associated signatures

  * OWA_deployment.xml?The deployment file

  * OWA_NS10_what is new.txt?A brief list of changes to the template since the previous version

  After you extract the template archive, in the AppExpert pane click Import AppExpert Template to run the AppExpert

wizard, and follow the instructions in the Wizard to install the template and create the OWA configuration.

For more information, see http://support.citrix.com/proddocs/topic/netscaler-10-1/ns-appexpert-con-10.html.

[From Build 112.15] [#246845]

- Configuring SourceIP for AppFlow Traffic

  You can now configure the source IP address (SNIP or MIP address), to be used for AppFlow traffic. When you add an Appflow collector by using the add appflow collector command, you can use the -netprofile option to associate a netprofile to which the source IP address is bound. By default, the Appflow exporter takes NSIP address as the source IP address if you do not specify the -netprofile option.

  > add appflow collector <col_name> -IPAddress <IP_addr> [-netprofile {netprofile_name}]

  [From Build 112.15] [#288343]

- Export Multiple Set-cookies in AppFlow Records

  The HTTP response can contain multiple values in the set-cookie header. This enhancement extends support to export all those values in the appflow record instead of just one value as was the case earlier.

  [From Build 112.15] [#329122]

- X-Forwarded-For HTTP Header Support

  AppFlow records can now log X-Forwarded-For HTTP header information. You can enable the logging with the "set appflow param -httpXForwardedFor ENABLED" command or by using the configuration utility.

  [From Build 112.15] [#311033]

- NetScaler Insight Center appliances now support exporting ICA AppFlow records from NetScaler appliances with enterprise licenses.

  [From Build 119.7] [#395659]

- Application-level Quality of Experience (AppQoE) integrates several existing policy-based security features of the NetScaler appliance into a single integrated feature that takes advantage of a new queuing mechanism, fair queuing. Fair queuing manages requests to load-balanced web servers and applications at the virtual server level instead of at the service level, allowing it to handle queuing of all requests to a web site or application as one group before load balancing, instead of as separate streams after load balancing. The integrated features are:

  - HTTP Denial-of-Service Protection (HDOSP)

  - Priority Queuing

  - SureConnect

  By implementing these features at the virtual server level instead of the individual service level, the NetScaler appliance can maintain absolute priority of connections, prevent flushing of connection if a service transitions state, and detect

and divert unwanted or lower priority traffic during DDoS attacks or other periods of extremely high load without having to first expend CPU to load balance these unwanted connections and assign them to a service queue.

For more information about AppQoE and instructions on how to implement it, see http://support.citrix.com/proddocs/topic/ns-main-appexpert-10-1-map/ns-appqoe-wrapper-con.html.

[From Build 112.15] [#379091]

- HTML Cross-Site Scripting Check Might Transform Allowed Tags and Attributes

  If an application firewall profile has the HTML cross-site scripting check configured to transform unsafe HTML, in some situations the application firewall might transform all HTML tags, including allowed HTML tags and attributes.

  [From Build 112.15] [#369529]

- Application Firewall Improved Diagnostics and Tracking Tools for Troubleshooting

  The application firewall now generates log messages for system resets, packets dropped because of violations of RFC strict checks or malformed request/response header checks, or due to errors within the application firewall itself. These logs provide additional information for troubleshooting.

  [From Build 112.15] [#248186]

- Application Firewall Cluster Support

  Support for the application firewall has been added to the NetScaler cluster when operated in single node spotted VIP mode. Application firewall commands run correctly at the command line, and the configuration utility displays the application firewall node and screens. Users should keep in mind that session state sharing between nodes is disabled when using the application firewall on a cluster.

  [From Build 112.15] [#326635]

- Application Firewall Learning Support on Cluster

  Support for the application firewall learning feature has been added to the NetScaler cluster. The cluster controller node now aggregates learning data from all nodes in the cluster and stores the learned data in a temporary database file. It then provides the data set to each node in the cluster upon request, enabling the learning feature to operate on the complete set of requests and responses to a protected web server, application, or service.

  [From Build 112.15] [#327601, 315156, 318640]

- Application Firewall Performance Improvements

  A number of performance improvements have increased the performance of the application firewall overall by approximately 10%. These improvements include caching of frequently used objects, significant enhancements to processing of HTTP POST bodies, and more efficient Signatures string operations.

  [From Build 112.15] [#327608, 206010]

- Application Firewall Scan Tool Integration

  The Citrix NetScaler Application Firewall now supports signatures generated by the IBM AppScan, Trend Microsystems,

and WhiteHat vulnerability scanners. You can import WhiteHat WASC 1.0, WASC 2.0, and best practices signatures, IBM AppScan Standard and Enterprise signatures, and Trend Microsystems Vulnerability Scanner (TMVS) signatures into the application firewall. These signatures can either be added to existing signatures objects, or can be used to create new signatures objects. Once imported, the signatures can be used to protect web applications exactly like any other signatures.

Once imported, the signatures can be used to protect web applications exactly like any other signatures.

For more information, see http://support.citrix.com/proddocs/topic/ns-security-10-1-map/appfw-signatures-con.html and http://support.citrix.com/proddocs/topic/ns-security-10-1-map/appfw-signatures-updatingcenzic-tsk.html.

[From Build 112.15] [#317580]

- The Citrix NetScaler Application Firewall Signatures feature has received a number of enhancements. The Signatures feature now includes the following new and enhanced functions:

  - Automatic updates: You can configure automatic updates for the default application firewall signatures or any signatures object that you have created from a cloud-based service. This feature is disabled by default. You enable and configure it in the configuration utility Signatures pane by selecting the signatures that you want to update, then choosing Auto-Update Settings in the Action drop-down list. If signature updates are enabled, the NetScaler appliance checks the specified URL for updates at the designated interval, hourly by default. If it finds updated signatures, it downloads and installs them.

  - Manual per-signature updates: Manual per-signature updates--You can manually update the default application firewall signatures or any signatures object that you have created by using the command line or the configuration utility. To update signatures from the command line, use the following command: update appfw signatures <name> [-mergeDefault].

  For <name>, substitute the name of signatures object to update. If you want to merge updates with the default signatures, include the -mergeDefault parameter.

  To update signatures by using the configuration utility, in the Signatures pane select the signatures that you want to update, then select Merge from the Action drop-down list. In the Update Signatures Object dialog box, type in the path and name of the signatures update file or use the browse dialog to select it, and then click Update.

  - Signature patterns support for JSON payloads: The signatures feature now matches JSON in HTTP requests. You can create patterns that examine JSON payloads for patterns that might signify a security breach on your protected web server or application.

  - Signature patterns support for HTTP responses: The signatures feature now matches patterns in the HTTP response as well as the request. You can create patterns that examine HTTP response headers and bodies for patterns that might signify a security breach on your protected web server or application.

  The following new patterns apply specifically to responses:

  Credit cards

  Safe objects

  - Per-signature counters: Signature statistics are now maintained on a per-signature basis, allowing you to see exactly how many times a specific signature has matched a request or response.

For more information about enhanced Signatures features, see http://support.citrix.com/proddocs/topic/ns-security-10-1-map/appfw-signatures-con.html.

[From Build 112.15] [#318148]

- When configuring the Safe Commerce (credit card) check, you can now configure the application firewall to check the MIME/type of HTTP responses and skip responses that are not of the appropriate content type for Safe Commerce filtering. You can use this configuration option to prevent false positives.

  To enable MIME/type checking, at the NetScaler command line type the following command:

  bind appfw profile <name> -inspectResContentType <type>

  For <name>, substitute the name of the profile. For <type>, substitute a string that matches the MIME/type. For example, to check for and skip PDF content sent to the library profile, you would type the following:

  bind appfw profile library -inspectResContentType "text/PDF"

  To disable a MIME/type rule that you have previously enabled, use the unbind command:

  unbind appfw profile <name> -inspectResContentType <type>

  [From Build 119.7] [#236218, 213852]

- Cache Redirection Changes

  The following changes have been made in the cache redirection feature:

  * The cacheVserver parameter is no longer part of the add cr vserver command. To specify a cache server, you must use the bind cr vserver ?lbvserver <string> command.

  In the configuration utility, in the Create Virtual Server (Cache Redirection) and Configure Virtual Server (Cache Redirection) dialog boxes, the Cache Server list has been renamed to Default Cache Server and has been moved from the Advanced tab to the area above the tabs. Additionally, a hit counter has been added next to the list. The hit counter maintains a count of the number of hits received by the cache server.

  * In the Create Virtual Server (Cache Redirection) and Configure Virtual Server (Cache Redirection) dialog boxes, on the Policies tab, when you click the CSW button and then the Insert Policy button, the list that appears in the Policy Name column no longer includes a Default content switching policy.

  [From Build 112.15] [#319966, 325728, 330010]

- You can now bind compression and filter policies to a cache redirection virtual server by using the configuration utility.

  [From Build 112.15] [#330033]

- AutoScale: Automatically Scaling Your Application Fleet in a CloudPlatform Environment

  Issue ID 0311703: In an environment deployed and managed by using Citrix CloudPlatform, automatic scaling of an application fleet can be achieved by using the Citrix NetScaler appliance. CloudPlatform provides a feature called

AutoScale, as part of its elastic load balancing feature. A CloudPlatform user can use the AutoScale feature to specify thresholds for various conditions for automatically scaling the application fleet upward and downward. The scale up and scale down conditions can vary from simple use cases, such as a server's CPU usage, to complex use cases, such as a combination of a server's CPU usage and responsiveness. CloudPlatform, in turn, configures the NetScaler appliance to load balance traffic to the application virtual machines (VMs), monitor application thresholds and performance, and trigger scale up and scale down actions to add or remove VMs from the application fleet.

For more information about how AutoScale works on the NetScaler appliance, see http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-autoscale-automatic-scaling-in-cloudplatform-env-wrapper-con.html.

For answers to frequently asked questions, see http://support.citrix.com/proddocs/topic/ns-faq-map-10-1/ns-faq-autoscale-ref.html.

[From Build 112.15] [#311703, 326608]

- The NetScaler cluster now supports the rate limiting and action analytics feature.

  [From Build 112.15] [#341764]

- The NetScaler cluster now supports configuring of content switching actions.

  [From Build 112.15] [#317324]

- Removing a Cluster Node

  You can now remove a cluster node through a single-step procedure. You must log on to the cluster IP address and execute the "rm cluster node" command.

  For more information, see http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-cluster-remove-node-tsk.html.

  [From Build 112.15] [#291771]

- Partially Striped Configurations in a NetScaler Cluster

  You can now define some configurations to be active only on specific cluster nodes. For example, you can define a virtual server to be active on only three nodes of a 5-node cluster. Such a configuration is referred to as partially striped. To define a partially striped configuration, use a node group, which is a set of cluster nodes to which you can bind the following virtual servers (load balancing, content switching, cache redirection, and authentication).

  Note: An entity that is bound to a node group that includes all the cluster nodes is striped across the cluster. Similarly, an entity that is bound to a node group that includes only one node is spotted on that node.

  For more information, see http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-cluster-node-groups-con.html and http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-cluster-config-node-group-tsk.html.

  [From Build 112.15] [#335401]

- The NetScaler cluster now supports spillover based on bandwidth.

[From Build 112.15] [#346786]

- The NetScaler cluster now supports Branch Repeater load balancing.

  [From Build 112.15] [#283450]

- Viewing and Clearing Node-Specific Routing Information

  You can now retrieve node-specific routing configurations by specifying the node(s) in the owner-node argument as follows:

  > vtysh

  ns# owner-node 0 1

  ns(node-0 1)# show cluster state

  ns(node-0 1)# exit-owner-node

  Similarly, you can also clear node-specific routing configurations by specifying the node(s) in the owner-node argument as follows:

  > vtysh

  ns# owner-node 0 1

  ns(node-0 1)# clear config

  ns(node-0 1)# exit-owner-node

  [From Build 112.15] [#309178]

- The NetScaler cluster now supports the ISIS routing protocol.

  [From Build 112.15] [#274535]

- The NetScaler cluster now supports IP-IP Tunneling.

  [From Build 112.15] [#269113]

- Configuring priority for the configuration coordinator

  You can now configure the priority for a cluster node to be selected as a configuration coordinator. The node with the highest priority (lowest priority number) is made the configuration coordinator. If the current configuration coordinator goes down, the node with the next lowest priority number takes over as the configuration coordinator. If the priority is not set or if there are multiple nodes with the lowest priority number, the configuration coordinator is selected from one of the available nodes.

  You can set the node priority by using the priority parameter of the add cluster node command.

  [From Build 112.15] [#359806]

- In addition to the reorganization of the nodes within the navigation tree, some of the nodes are now grouped with the

configurations options in the details pane (the pane on the right side of the screen) of the configuration utility. For example, LDNS entries, which were a subnode of GSLB, are now with the global GSLB configuration items in the details pane.

The following embedded Java views have been moved to the Overview pages:

- Auto Detected Services Detail View

- FIPS Detail View

- Applications Detail View

- Access Gateway Applications Detail view

- Template Detail view

- GSLB LDNS entries Detail View

- Cache Objects

[From Build 112.15] [#381622]

- Features in the NetScaler configuration utility navigation tree have been reorganized to provide greater logical consistency and ease of navigation. The feature nodes are grouped under the following top-level nodes:

  - System: System and infrastructure features

  - AppExpert: Grouping of all Application, Policies, templates and Layer 7 features

  - Traffic Management: Core traffic management features such as load balancing, GSLB, content switching, cache redirection, SSL, and SSL offload

  - Optimization: Core optimization features such as caching and compression

  - Security: Security oriented features and functionalities

  For more information, see http://support.citrix.com/proddocs/topic/ns-rn-main-release-10-1-map/ns-rn-changes-gui-10-1-con.html.

  [From Build 112.15] [#360658]

- Global Setting for Using a Proxy Port

  You can now use the NetScaler user interface to configure the Use Proxy Port setting globally.

  For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-advancedsettings-useproxyport-tsk.html.

  [From Build 112.15] [#302646]

- The content switching feature now supports dynamic selection of a load balancing virtual server at the run time. This feature enables you to analyze the request and accordingly direct it to the correct load balancing virtual server. The target LB virtual server is determined at the run time by the expression defined in the action of the content switching

policy.

[From Build 112.15] [#248750]

- Rename a Content Switching Policy Label

  You can now rename a content switching policy label, even if the label is already referenced by existing policies. The new name is automatically incorporated into all configurations that include the old name.

  For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-cs-basicconfig-policy-labels-tsk.html.

  [From Build 112.15] [#312929]

- Enabling or Disabling the Recursion Available Flag

  An option Recursion Available is added for the load balancing virtual servers of type DNS and DNS TCP to control the RA (Recursion Available) flag in all the DNS responses from these virtual servers.

  [From Build 119.7] [#403114, 248936, 269857, 388338]

- The NetScaler DNS64 feature responds with a synthesized DNS AAAA record to an IPv6 client sending an AAAA request for an IPv4-only domain. The DNS64 feature is used with the NAT64 feature to enable seamless communication between IPv6-only clients and IPv4-only servers. DNS64 enables discovery of the IPv4 domain by the IPV6 only clients, and NAT64 enables communication between the clients and servers.

  For synthesizing an AAAA record, the NetScaler appliance fetches a DNS A record from a DNS server. The DNS64 prefix is a 96-bit IPv6 prefix configured on the NetScaler appliance. The NetScaler appliance synthesizes the AAAA record by concatenation of the DNS64 Prefix (96 bits) and the IPv4 address (32 bits).

  For more information on configuring DNS64, see http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-nw-ipaddrssng-DNS64-intro-con.html.

  [From Build 120.13] [#318404]

- Database Profiles

  You can now configure a database profile for virtual servers of type MSSQL and MYSQL. A database profile is a named collection of parameters that is configured once but applied to multiple virtual servers that require those particular parameter settings. After creating a database profile, you bind it to load balancing or content switching virtual servers. You can create as many profiles as you need.

  For more information, see http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-ac-config-db-profile-tsk.html.

  [From Build 112.15] [#343179]

- Transparent Mode for Logging MSSQL Transactions

You can configure the NetScaler appliance to operate transparently between MSSQL clients and servers, and to only log or analyze details of all client-server transactions. Transparent mode is designed so that the NetScaler appliance only forwards MSSQL requests to the server, and then relays the server's responses to the clients. As the requests and responses pass through the appliance, the appliance logs information gathered from them, as specified by the AppFlow configuration, or collects statistics, as specified by the Action Analytics configuration.

For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-dbproxy-usecase-log-mssql-transactions-transparent-mode-tsk.html.

[From Build 112.15] [#319464]

- Database Specific Load Balancing of Services

  You can now configure the Citrix NetScaler appliance to retrieve a list of databases that are active on a service and, for a given query, to load balance only the services on which the requested database is available. If the requested database is unavailable on a service, the appliance excludes the service from load balancing decisions until it becomes available. This behavior ensures uninterrupted service to clients.

  For more information about database specific load balancing, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-dbproxy-db-specific-lb-wrapper-con.html.

  [From Build 112.15] [#358254]

- Support for Microsoft SQL Server 2012

  The Citrix NetScaler appliance now supports Microsoft SQL Server 2012. To load balance SQL 2012 database servers, you must set the Server Version (mssqlServerVersion) parameter to 2012 on each of the load balancing and content switching virtual servers in the configuration.

  If you have configured availability groups for read-only routing, the appliance can handle the redirect packets with which the primary database server responds to clients who declare read-only application intent in their connection properties. However, when deployed to manage traffic associated with an availability group, the NetScaler appliance provides additional benefits. With the help of content switching policies, the appliance can differentiate between connections in which the ApplicationIntent connection property is set to ReadWrite and those in which the property is set to ReadOnly. A content switching virtual server can then forward all ReadWrite requests to a load balancing virtual server to which you have bound the primary database instance, and all ReadOnly requests to a load balancing virtual server to which you have bound the secondary database servers.

  In this configuration, ReadOnly requests are load balanced across all the secondary servers (unlike configurations involving a redirect response, in which only one secondary server is selected for serving ReadOnly requests). In this way, the appliance can optimally utilize all of the secondary database servers while eliminating redirect traffic from your network.

  [From Build 112.15] [#354723]

- Caching Stored Procedures and SQL Queries

  If connection multiplexing is disabled in a database profile, stored procedures and SQL batch queries are not cached, despite caching being enabled for the profile. With this enhancement, you can enable caching, if connection multiplexing is disabled, by setting the new "enableCachingConMuxOFF" parameter in the profile.

At the command prompt, type:

add dbProfile <name> ?conMultiplex DISABLED -enableCachingConMuxOFF ENABLED

or

set dbProfile <name> -enableCachingConMuxOFF ENABLED

In the configuration utility, select "Enable caching when connection multiplexing OFF".

[From Build 126.12] [#453973]

- View Site Persistence Cookies for GSLB Services

  If site persistence is configured for GSLB services, and the services are bound to a GSLB virtual server, the NetScaler appliance generates a site persistence cookie for each service. Unlike in earlier NetScaler releases, the NetScaler user interface now displays the site persistence cookies that the appliance generates.

  To view site persistence cookies by using the NetScaler command line

  At the NetScaler command prompt, type:

  show gslb vserver <name>

  To view site persistence cookies by using the NetScaler configuration utility

  1. In the navigation pane, expand GSLB, and then click Virtual Servers.

  2. In the details pane, select the virtual server for whose services you want to view site persistence cookies, and then click Open.

  3. In the Configure GSLB Virtual Server dialog box, on the Services tab, select the service whose site persistence cookie you want to view.

  The site persistence cookie is displayed below the table of services.

  [From Build 112.15] [#242446]

- Diameter Expression Support

  Expressions to retrieve AVPs from a Diameter request or response are now available. You can use these expressions for configuring the token load balancing method and for rule-based persistency.

  The expressions are of the form DIAMETER.REQ.AVP(<avpcode>). For example, to retrieve the Auth-Application-Id AVP (AVP code 258), you can use the expression: DIAMETER.REQ.AVP(258).

  Some important AVPs have aliases. For example, the Auth-Application_Id AVP has the alias AUTH_APPLICATION_ID. So, the expression to retrieve the Auth-Application_Id by using the alias is: DIAMETER.REQ.AUTH_APPLICATION_ID.

  [From Build 112.15] [#318377]

- Rate Statistics for Services Bound to a Load Balancing Virtual Server

  The stat lb vserver command and the Monitoring page for a load balancing virtual server now display the hit rate (Hits/s), request rate(Req/s), and response rate (Rsp/s) for bound services.

  [From Build 112.15] [#275029]

- Automatic State Transition Based on Percentage Health of Bound Services

  You can now configure a load balancing virtual server to automatically transition from the UP state to the DOWN state if the percentage of active services falls below a configured threshold. For example, if you bind 10 services to a load balancing virtual server and configure a threshold of 50% for that virtual server, it transitions from UP to DOWN if six or more services are DOWN. When the percentage health rises above the threshold value, the virtual server returns to the UP state. You can also enable an SNMP alarm called ENTITY-STATE if you want the NetScaler appliance to notify you when the percentage health of bound services causes a virtual server to change state.

  For instructions, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-advancedsettings-auto-state-transition-svc-health-tsk.html.

  [From Build 112.15] [#361659]

- Stateless Connection Failover Supported for IPv6

  You can now bind an IPv6 service to a load balancing virtual server with connection failover set to stateless.

  [From Build 112.15] [#276300]

- Configure Spillover Based on NetScaler Policies

  In earlier NetScaler releases, you can configure spillover by specifying only one of the following spillover methods along with a spillover threshold: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, and HEALTH. Also, of a backup virtual server is not available when spillover occurs, the NetScaler appliance responds to clients with a TCP reset.

  In this release, you can also use a NetScaler rule, of your choice, to specify the conditions that should be met for spillover to occur. You specify the rule in a spillover policy. Configuring a spillover rule enables you to configure the NetScaler appliance for a wider range of spillover scenarios. For example, you can configure spillover on the basis of the virtual server's response time, or on the basis of the load on the virtual server.

  [From Build 112.15] [#257226]

- Monitor for Citrix StoreFront Stores

  You can now configure a user monitor for a Citrix Storefront store.

  For more information about monitoring a StoreFront store, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-monitors-builtin-ctx-storefront-stores-tsk.html.

  [From Build 112.15] [#366050]

- Ability to Specify a Name for a Persistence Cookie

  Unlike in earlier releases, for load balancing virtual servers and load balancing persistency groups for which the COOKIEINSERT persistence type is configured, you can specify a name for the persistency cookie. You specify a name

for the persistency cookie by setting the cookieName parameter. If you configure the COOKIEINSERT persistence type, but you do not specify the cookieName parameter, the NetScaler appliance inserts a cookie of the form <NSC_XXXX>= <serviceIP> <servicePort>, where <NSC_XXXX> is the virtual-server ID that is derived from the virtual server's name, <serviceIP> is the hexadecimal value of the IP address of the service, and <servicePort> is the hexadecimal value of the port of the service.

[From Build 112.15] [#289773, 232227, 289772, 302494]

- Counters for the Number of Active and Inactive Bound Services

  Issue ID 0275028: The stat lb vserver and stat gslb vserver commands, and the Monitoring pages for load balancing and global server load balancing virtual servers, now display a count of the number of bound services that are UP and DOWN. The counters are called actSvcs (total active services) and inactSvcs (total inactive services), respectively.

  [From Build 112.15] [#275028]

- View the Global Spillover Count by Using SNMP

  You can use the totSpilloverCount SNMP counter to retrieve a count of the number of times spillover has occurred on various load balancing and content switching virtual servers after the NetScaler appliance was last restarted. The SNMP OID is 1.3.6.1.4.1.5951.4.1.3.5.6.

  [From Build 112.15] [#229026]

- Support for Clearing a Specific Persistence Session

  Issue ID 0258312: You can specify a persistence parameter in the "clear lb persistentSessions" command to clear the persistence session associated with only that parameter. Following is the command synopsis for clearing the session associated with a specific persistence parameter:

  clear lb persistentSessions [<vServer> [-persistenceParam <string>]]

  where

  persistenceParam is the persistence parameter whose session you want to clear.

  For more information about clearing a specific persistence session, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-persistence-clearing-tsk.html.

  [From Build 112.15] [#258312]

- Ability to Configure VLAN Transparency

  You can now configure a load balancing virtual server to retain the client's VLAN identifier in packets that are to be forwarded to servers. The virtual server must be a wildcard virtual server of type ANY, and must be functioning in MAC mode.

  For instructions, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-advancedsettings-retain-vlan-tsk.html.

  [From Build 112.15] [#361552]

- Offload DNSSEC Operations to the NetScaler Appliance

For DNS zones for which your DNS servers are authoritative, you can offload DNSSEC operations to the NetScaler appliance. When a DNS server sends a response, the appliance signs the response on the fly before relaying it to the client. The appliance also caches the signed response. Apart from reducing the load on the DNS servers, offloading DNSSEC operations to the appliance gives you the following benefits:

-> You can sign records that the DNS servers generate programmatically. Such records cannot be signed by routine zone signing operations performed on the DNS servers.

-> You can serve signed responses to clients even if you have not implemented DNSSEC on your servers.

[From Build 112.15] [#249691]

- Support for Overriding Persistence for Overloaded Services

  When a service is loaded or is otherwise unavailable, service to clients is degraded. To work around this situation, you might have to configure the NetScaler appliance to temporarily forward to other services the requests that would otherwise be included in the persistence session that is associated with the overloaded service. In other words, you might have to override the persistence setting that is configured for the load balancing virtual server until the service returns to a state in which it can accept requests. You can achieve this functionality by binding a load monitor to the virtual server and setting the skippersistency parameter for the virtual server.

  For more information about overriding persistence for overloaded services, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-persistence-override-pers-overloaded-server-tsk.html.

  [From Build 112.15] [#258313]

- Increase in the Maximum Number of Persistence Sessions

  The maximum number of persistence sessions per core on an nCore NetScaler appliance has been raised from 150,000 to 1,000,000 (1 million). The maximum number of persistence sessions that can coexist on an nCore NetScaler appliance is equal to the product of the number of cores and the per-core limit. For example, if the appliance has 6 CPU cores, the maximum number of persistence sessions that can coexist on the appliance is 6,000,000 (6 * 1000000).

  For information about how to configure a limit for the number of persistence sessions that can coexist on the NetScaler appliance, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-persistence-config-limit-number-persist-sessions-tsk.html.

  [From Build 112.15] [#328498]

- Monitor for Accounting Information Delivery from a RADIUS Server

  You can now configure a monitor called a RADIUS accounting monitor to determine whether the Radius server used for Authentication, Authorization, and Accounting (AAA) is delivering accounting information as expected.

  For more information about monitoring accounting information delivery from a RADIUS server, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-1-map/ns-lb-monitors-builtin-radius-accnting-tsk.html.

  [From Build 112.15] [#348828]

- Dynamic Load Balancing of Repeater Appliances

You can now configure the NetScaler appliance for dynamic load balancing of Repeater appliances, by using the Dynamic Load Balancing wizard for Citrix Branch Repeater. In the wizard, you specify the datacenter Repeater IP addresses and the datacenter server subnets to which the NetScaler appliance or instance must forward the branch-office traffic. The wizard creates the required configuration.

[From Build 112.15] [#333238]

- Options for Branch IP Address in the Load Balancing wizard for Citrix Branch Repeater

  In the Static Load Balancing wizard for Citrix Branch Repeater, when specifying a branch whose traffic is to be accelerated, you can specify either the primary IP address or the accelerated pair A (apA) IP address of a Branch Repeater appliance.

  [From Build 112.15] [#275289]

- NetScaler and XenMobile Solution for Enterprise Mobility

  Citrix NetScaler deployed with XenMobile Mobile Device Management (MDM) provides the ability to scale, ensure high availability for apps, and maintain security.

  Use the XenMobile MDM Setup wizard on the NetScaler configuration utility to configure the following two deployment scenarios:

  * Load balance XenMobile Device Managers (MDM servers): In this scenario, the NetScaler appliance sits between the client and the XenMobile MDM servers to load balance encrypted data from mobile devices to the XDM servers.

  * Load balance MS Exchange servers with email filtering: In this scenario, the NetScaler appliance sits between the client and the XNC and CAS servers. All requests from the client devices go to the NetScaler appliance, which then communicates with the XNC to retrieve information about the device. Based on the response from the XNC, the NetScaler either forwards the request from a whitelisted device to the backend server, or drops the connection from a blacklisted device.

  [From Build 118.7] [#365382]

- Oracle Monitor Support

  You can now create a load balancing monitor for an Oracle DBMS server by using the new Oracle-ECV monitor type. This monitor supports the following data types:

  ORACLE_BINARY_DOUBLE = 101,

  ORACLE_BINARY_FLOAT = 100,

  ORACLE_CHAR = 96,

  ORACLE_DATE = 12,

  ORACLE_INTERVALDS = 183,

  ORACLE_INTERVALYM = 182,

  ORACLE_NUMBER1 = 2,

ORACLE_NUMBER2 = 6,

ORACLE_NVARCHAR2 = 1,

ORACLE_TIMESTAMP = 180,

ORACLE_TIMESTAMP_WITH_LOCAL_TIME_ZONE = 231

ORACLE_TIMESTAMP_WITH_TIME_ZONE = 181,

You can configure the monitor by using the NetScaler command line or the configuration utility. To create or configure an Oracle-ECV monitor at the NetScaler command line, type the appropriate command:

add lb monitor <monitorName> oracle-ecv [parameters...]

set lb monitor <monitorName> oracle-ecv [parameters...]

To create or configure an Oracle-ECV monitor by using the configuration utility, navigate to Traffic Management => Load Balancing => Monitors, and then click Add to create the monitor or select an existing monitor and then click Open to configure the monitor.

The new expressions that support the Oracle-ECV monitor are as follows:

ORACLE.RES.ATLEAST_ROWS_COUNT(n)

Determines whether the query response contains at least the specified

number of rows.

ORACLE.RES.ROW(i).NUM_ELEM(j).eq(n)

Determines whether the value located at the specified row and column

is equal to the specified number. You can substitute other valid

numeric operations for "eq".

ORACLE.RES.ROW(i).IS_NULL_ELEM(j)

Determines whether the value located at the specified row and column

is NULL.

ORACLE.RES.ROW(i).TEXT_ELEM(j).eq("pattern")

Determines whether the value located at the specified row and column

matches the specified pattern. You can substitute other valid

text operations for "eq".

[From Build 118.7] [#364085]

- Setting Up NetScaler for XenApp/XenDesktop

  The NetScaler now provides a wizard that simplifies the task of setting up a NetScaler appliance for a

XenApp/XenDesktop deployment. For more information, see Setting Up NetScaler for XenApp/XenDesktop.

[From Build 120.13] [#345912]

- You can now configure up to 8K (8192) service groups on a NetScaler appliance. The earlier limit was 4K (4096) service groups.

  [From Build 121.10] [#406355]

- Native Windows Authentication (Kerberos) for MSSQL Monitors

  Microsoft SQL monitors on the NetScaler appliance now support the Kerberos authentication protocol, and can therefore monitor load-balanced application servers in a Kerberos 5 environment that employs Kerberos Protocol Transition (KPT) and Kerberos Constrained Delegation (KCD).

  [From Build 112.15] [#329542]

- Starting with release 10.1 build 122.17, the script files for user monitors are in a new location.

  If you upgrade an MPX or VPX virtual appliance to release 10.1 build 122.17 or later, the changes are as follows:

  - A new directory named conflicts is created in /nsconfig/monitors/ and all the built-in scripts of the previous builds are moved to this directory.

  - All new built-in scripts are available in the /netscaler/monitors/ directory. All custom scripts are available in the /nsconfig/monitors/ directory.

  - You must save a new custom script in the /nsconfig/monitors/ directory.

  - After the upgrade is completed, if a custom script is created and saved in the /nsconfig/monitors/ directory, with the same name as that of a built-in script, the script in the /netscaler/monitors/ directory takes priority. That is, the custom script does not run.

  If you provision a virtual appliance with release 10.1 build 122.17 or later, the changes are as follows:

  - All built-in scripts are available in the /netscaler/monitors/ directory.

  - The /nsconfig/monitors/ directory is empty.

  - If you create a new custom script, you must save it in the /nsconfig/monitors/ directory.

  [From Build 122.17] [#447105]

- Unlicensed Feature Handling

  NITRO operations are now restricted to the features that are licensed on the NetScaler appliance.

  [From Build 112.15] [#328055]

- You can now view the virtual servers to which a specified service is bound. The REST URL for this is http://<nsip>/nitro/v1/config/svcbindings/svcname.

  [From Build 112.15] [#257279]

- Log Support

  All NITRO operations are now logged in the /var/nitro/nitro.log file on the appliance.

  [From Build 112.15] [#328051]

- Plug-in Icon Decoupling from Citrix Receiver

  The desktop client plug-ins icons can now be configured to operate independently from Native Citrix Receiver clients. Settings to manage Receiver integration with the NetScaler Gateway Plug-ins can be configured globally and within session policies.

  [From Build 129.22] [#406312]

- The HTML Injection feature is now available for Web Insight data collection on platinum licenses of NetScaler 10.0 appliances and on all licenses of NetScaler 10.1 appliances.

  [From Build 118.7] [#392732]

- On the Dashboard > HDX Insight > Users > <user name> page, the application and gateway reports display the active applications by default.

  [From Build 118.7] [#388409]

- NetScaler Insight Center supports clearing AppFlow configurations from a virtual server.

  [From Build 118.7] [#341904, 375905, 383246]

- NetScaler Insight Center supports sending syslog messages to an external syslog server.

  [From Build 118.7] [#381072]

- HDX Insight reports now include details about Client Side NS Latency, Server Side NS Latency and Host Delay.

  [From Build 119.7] [#400867]

- NetScaler Insight Center now saves the following:

  Granular data: Time to Purge

  7 sec data: 6 min

  5 min data: 65 min.

  Hourly data: 25 hrs.

Daily data: 8 days.

Weekly data: 5 weeks.

[From Build 121.10] [#404805]

- HDX Insight reports now include details about session reconnects, client-side retransmissions, and server-side retransmissions.

  [From Build 122.17] [#392016]

- All the metrics except bandwidth and hits display the average values.

  [From Build 122.17] [#409634]

- HDX Insight now provides a report about active sessions, grouped by server IP and gateway IP.

  [From Build 122.17] [#398322]

- The top-right corner of the page now displays a percentile icon, which you can click to display percentile values and the highest and lowest values for a selected metric.

  [From Build 122.17] [#418196]

- In this release you can select and show columns in the tables on the NetScaler Insight Center graphical user interface (GUI) and also rearrange them. The changes can also be made persistent to reflect these changes when the same user logs in the next time.

  [From Build 122.17] [#423207]

- HDX Insight reports now include details about CloudBridge in an ICA session path.

  [From Build 123.11] [#432702, 430583]

- You can now configure the ICA session timeout value for inactive sessions on the configuration tab of the NetScaler Insight Center.

  For details, see http://support.citrix.com/proddocs/topic/ni-10-5-map/ni-ica-session-timeout-tsk.html

  [From Build 123.11] [#431957]

- Configuring VLANs on Management Interfaces

  You can now configure a VLAN on the management interfaces, 0/1 and 0/2, while provisioning a NetScaler instance.

  [From Build 112.15] [#318609]

- Support for SNMPv3 Queries on the NetScaler SDX Appliance

  Simple Network Management Protocol Version 3 (SNMPv3) queries are now supported on the NetScaler SDX appliance. SNMPv3 enhances the basic architecture of SNMPv1 and SNMPv2 to incorporate administration and security capabilities, such as authentication, access control, data integrity check, data origin verification, message timeliness check, and data confidentiality.

For more information, see http://support.citrix.com/proddocs/topic/sdx-administration-10-1-map/sdx-ag-manage-mon-appliance-config-snmp-v3-con.html.

[From Build 112.15] [#328392]

- Support of Regular Expressions for Search Text Fields

  The Search text fields on the pagination views of the Management Service utility now support regular expressions.

  [From Build 112.15] [#309358, 312469]

- Changing the Hostname of the Appliance

  You can now change the hostname of the Management Service. On the Configuration tab, navigate to System > System Settings > Change Hostname, and enter a new hostname.

  [From Build 112.15] [#323534]

- Password Management on the NetScaler SDX Appliance

  If you log on to a NetScaler VPX instance and change the password for access to the instance, instead of changing the password from the Management Service, connectivity from the Management Service to the instance is lost. With this release, you can restore connectivity by creating a new profile from the Management Service, assigning it the same password that you specified on the NetScaler VPX instance, and then binding the new profile to the NetScaler VPX instance.

  [From Build 112.15] [#318968]

- Support for System Notifications

  You can now configure Syslog, mail, and SMS notifications on the SDX appliance.

  [From Build 112.15] [#291016]

- Restrict a VLAN to a Specific Virtual Interface

  The NetScaler SDX appliance administrator can enforce specific 802.1Q VLANs on the virtual interfaces associated with NetScaler instances. This capability is especially helpful in restricting the usage of 802.1Q VLANs by the instance administrators. If two instances belonging to two different companies are hosted on an SDX appliance, you can restrict the two companies from using the same VLAN ID, so that one company does not see the other company's traffic. If an instance administrator, while provisioning or modifying a VPX instance, tries to assign an interface to an 802.1Q VLAN, a validation is performed to verify that the VLAN ID specified is part of the allowed list.

  For more information, see http://support.citrix.com/proddocs/topic/sdx-administration-10-1-map/sdx-ag-prov-ns-instances-restrict-vlans-to-vfs-tsk.html.

  [From Build 112.15] [#323926]

- Audit Templates for NetScaler Instances

  You can create an audit template by copying the commands from an existing configuration file. You can later use this template to find any changes in the configuration of an instance and take corrective action if required.

[From Build 112.15] [#322404]

- Simplification of NetScaler SDX Licensing Process

  The process of allocating your licenses has been greatly simplified. The new licensing framework allows you to focus on getting maximum value from Citrix products.

  In the Management Service configuration utility (GUI), you can use your hardware serial number (HSN) or your license activation code (LAC) to allocate your licenses. Alternatively, if a license is already present on your local computer, you can upload it to the appliance.

  [From Build 112.15] [#323681, 331160]

- Display the Mapping of virtual interfaces on the VPX instance to the physical interfaces on the NetScaler SDX Appliance

  If you log on to the NetScaler virtual instance, the configuration utility and the command line interface display the mapping of the virtual interfaces on the instance to the physical interfaces on the appliance.

  For more information, see http://support.citrix.com/proddocs/topic/sdx-administration-10-1-map/sdx-ag-interface-naming-tsk.html.

  [From Build 112.15] [#261346, 335910]

- Cluster of NetScaler Instances Provisioned on NetScaler SDX Appliances

  You can now create a cluster of NetScaler instances that are provisioned on the NetScaler SDX appliance. The instances can be available on the same SDX appliance or on any SDX appliance within the same subnet.

  For more information, see http://support.citrix.com/proddocs/topic/sdx-administration-10-1-map/sdx-setup-cluster-tsk.html.

  [From Build 112.15] [#317258, 286049, 345509]

- Retrieving Tech Support tar file of Instances from Management Service Utility

  Now you can generate support tar archive for instances running on SDX through the Management Service utility.

  [From Build 112.15] [#240391]

- MAC-Address Assignment by System Administrator

  If, while you are provisioning a NetScaler instance on an SDX appliance, XenServer internally assigns a MAC address to a virtual interface associated with that instance, the same MAC address might be assigned to a virtual interface associated with another instance on the same appliance or on another appliance. To prevent assignment of duplicate MAC addresses, you can enforce unique MAC addresses.

  For more information, see http://support.citrix.com/proddocs/topic/sdx-administration-10-1-map/sdx-ag-config-svm-assign-mac-addr-to-interface-tsk.html.

  [From Build 112.15] [#325507]

- Provisioning Third-Party Instances on a NetScaler SDX Appliance

  You can now provision the following third-party virtual machines (instances):

BlueCat DNS/DHCP Server?Provides a DNS, DHCP, and IP Address Management software solution for enterprises.

[From Build 118.7] [#349549]

- Provisioning Third-Party Instances on a NetScaler SDX Appliance

  You can now provision the following third-party virtual machines (instances):

  * SECUREMATRIX(R) GSB?Provides a highly secure password system that eliminates the need to carry any token devices.

  * Websense(R) Protector?Allows enterprises to deploy a data loss prevention (DLP) solution to protect sensitive enterprise information.

  [From Build 118.7] [#329072]

- Upgrading the XenServer Software

  You must upgrade the NetScaler SDX appliance to XenServer version 6.1.0 to enable functionality of some features, such as LACP and third-party virtual machines. The process of upgrading the XenServer software involves uploading the build file of the target build to the Management Service, and then upgrading the XenServer software.

  [From Build 118.7] [#322368]

- Configure Link Aggregation from the Management Service

  You can now configure link aggregation from the Management Service at the time of provisioning a NetScaler instance, or later by modifying an instance. An aggregated link is also known as a channel. The interfaces that form part of a channel are not listed in the Network Settings view shown when you add or modify a NetScaler instance. Instead of the interfaces, the channels are listed.

  [From Build 118.7] [#257892]

- Upgrade Progress

  When you upgrade a NetScaler VPX instance on an SDX appliance, a new window, Upgrade Progress, shows the status of the upgrade operation, including any error messages. This feature is also available for SecureMatrixGSB and Websense Protector virtual machines.

  [From Build 120.13] [#346988]

- Multi-interface Support for BlueCat DNS/DHCP Server Virtual Machines

  Management Service now supports assigning interfaces explicitly for high availability and service along with the management for BlueCat DNS/DHCP Server virtual machines.

  [From Build 122.17] [#413839]

- 22040/22060/22080/22100/22120 Platform

  The SDX 22040/22060/22080/22100/22120 platform now supports NetScaler release 10.1 build 122.x.

  For more information, see Citrix NetScaler SDX 22040, SDX 22060, SDX 22080, SDX 22100, and SDX 22120.

  [From Build 122.17] [#353415]

- When system sends any e-mail notification, it will contain host name along with IP address as sender.

  [From Build 129.22] [#464856]

- You do not require a separate license file to set up a cluster on an SDX appliance. Clustering support will be provided with a valid SDX Platform License.

  [From Build 129.22] [#492668]

- Support for NetScaler VPX Virtual Appliance on XenServer 6.2

  The NetScaler VPX virtual appliance now supports XenServer version 6.2 only on a non-SDX appliance. On the NetScaler SDX appliance, only the XenServer versions available for download on www.citrix.com under NetScaler downloads are supported. XenServer 6.1.1 is the latest supported version on the NetScaler SDX appliance.

  [From Build 122.17] [#439509]

- NetScaler VPX Setup for the Linux KVM Platform

  The Citrix NetScaler VPX can now be hosted on Kernel-based Virtualization Machine (KVM). NetScaler VPX runs as a virtual appliance on Linux-KVM server. You can set up the NetScaler VPX on this platform either through the graphical Virtual Machine Manager (Virt-Manager) application or the vrish program.

  The host Linux operating system must be installed on suitable hardware by using virtualization tools such as KVM Module and QEMU. The number of virtual machines (VMs) that can be deployed on the hypervisor depends on the application requirement and the chosen hardware. After you provision a NetScaler virtual appliance, you can add additional interfaces.

  For more information, see Installing NetScaler Virtual Appliances on Linux-KVM Platform.

  [From Build 123.11] [#344349]

- Block Fragmented Packets

  You can now configure the NetScaler appliance to drop any fragmented packets that it receives.

  This feature can be useful in the following cases:

  *To preventing security attacks based on fragmented packets

  *To accommodate a use case requiring that the NetScaler appliance accept no fragmented packets.

  To block any fragmented packet, enable the Dropipfragments (Drop IP Fragments) option in one of the following ways:

  *On the NetScaler command line, by running the set L3 param command.

  *In the configuration utility, by using the Configure Layer 3 Parameters dialog box (Network > Settings > Configure Layer 3 Parameters).

  [From Build 112.15] [#299298]

- IPv6 Protocol Compliance

  The appliance accepts all the ICMPv6 fragments of an ICMPv6 echo request that is destined to one of the NetScaler owned IPv6 address. The appliance also sends out all the ICMPv6 fragments of the corresponding ICMv6 echo response.

  [From Build 112.15] [#286580]

- Block Non-IP Packets

  You can configure the NetScaler appliance to drop any non-IP related packet that it receives. For example, you can drop ARP packets. This feature can be useful in the following cases:

  - To prevent security attacks based on non-IP traffic

  - To prevent very heavy non-IP traffic from affecting the performance of the appliance

  To block non-IP traffic, set the fwmode (Network firewall mode) parameter to FULL in one of the following ways:

  - On the NetScaler command line, by running the set ns config command.

  - In the configuration utility, by using the Configure Network firewall mode settings dialog box (System > Settings > Change Network firewall mode).

  [From Build 112.15] [#329548]

- Controlling the L2 Conn Behavior of Load Balancing Virtual Servers

  The set l4 parameter command has a new parameter, l2connMethod, for specifying the MAC address, channel number, and VLAN ID attributes for the L2 Conn option behavior in a virtual server

  [From Build 112.15] [#339846, 332695]

- TFTP Support

  The NetScaler appliance now supports communication between a client and a Trivial File Transfer Protocol (TFTP) server.

  TFTP is a simple form of file transfer protocol and is based on the UDP protocol. TFTP does not provide any security features and is generally used for automated transfer of configuration and boot files between devices in a private network. TFTP support on the NetScaler appliance is compliant with RFC 1350. A server listens on port 69 for any TFTP request.

  The following features are supported:

  Load balancing of TFTP servers?The NetScaler appliance can now load balance TFTP servers.

  INAT processing compliant to TFTP?When a request packet, with port 69 as the destination, received by the NetScaler appliance matches an INAT rule with TFTP option enabled, the appliance processes the request and the corresponding response as compliant with the TFTP protocol.

  RNAT processing compliant to TFTP?When a request packet generated by a server is destined to a TFTP server, and the packet matches an RNAT rule on the NetScaler appliance, the appliance's processing of the request and the corresponding response from the TFTP server is compliant with the TFTP protocol.

  [From Build 112.15] [#250958, 244142, 258928]

- Configure Traffic Domains

  Traffic domains are a way to segment network traffic for different applications. You can use traffic domains to create multiple isolated environments within a NetScaler appliance. An application belonging to a specific traffic domain communicates with entities and processes traffic within that domain. The traffic belonging to one traffic domain cannot cross the boundary of another traffic domain.

  The main benefits of using traffic domains on the NetScaler appliance are the following:

  * Use of duplicate IP addresses in a Network?Traffic domains allow you to assign the same IP address or network address to multiple devices on a network, or multiple entities on a NetScaler appliance, as long as each of the duplicate address belongs to a different traffic domain.

  * Use of Duplicate entities on the NetScaler appliance?Traffic domains allow you to use duplicate NetScaler feature entities on the appliance. You can create entities with the same settings as long as each entity is assigned to a separate traffic domain.

  * Multitenancy?Using traffic domains, you can provide hosting services for multiple customers by isolating each customer's type of application traffic within a defined address space on the network.

  [From Build 112.15] [#319241, 318309]

- IPv6 Protocol Compliance

  The NetScaler appliance in L3 mode can now send out periodic Router Advertisement (RA) messages from its advertising interfaces. The appliance also sends RA messages in response to valid solicitations messages. The outgoing RA messages sent by the NetScaler appliance are compliant with RFC 4861 for Neighbor Discovery protocol for IP version 6 (IPv6). The NetScaler appliance can also send redirect messages to inform an originating host of a better router for reaching a specific destination.

  [From Build 112.15] [#286578]

- Powering off an Interface

  Now, when you disable an interface or an LA channel, the NetScaler appliance powers off the interface or interfaces of the LA channel and sends a link-down message to the peer device to notify that the interface(s) are disabled.

  [From Build 112.15] [#338863]

- Clearing all Dynamic Routing Configurations

  You can now at once clear all the routing configurations, which you created by using the VTYSH shell.

  For clearing all the dynamic routing configurations, you run the clear config command in the Exec mode of the VTYSH shell. After clearing the configuration, you must run the write command in the VTYSH shell to save the changes.

  [From Build 112.15] [#285913]

- ACL Action in the ACL Log Messages

  Each ACL log entry now includes a field that displays the action set for the ACL. This field tells you whether the packet that hit the ACL was passed onto the NetScaler appliance or was dropped.

The field takes one of the following values:

- ALLOW: A packet that matches the conditions specified in the ACL and is passed onto the NetScaler appliance.

- DENY: A packet that matches the conditions specified in the ACL and is dropped.

Following are two sample log entries:

19) 01/23/2013:18:48:53 GMT Informational 0-PPE-0 : ACL ACL_PKT_LOG 212 0 : Source 10.102.56.26

--> Destination 10.102.56.40 - Protocol ICMP -Type 8 - Code 0 -TimeStamp 92612208(ms) - Hitcount 5 -

Hit Rule ACL1 - Action ALLOW - Data 08 00 51 ac 7e 73 00 5c 19 c4 ff 50 19 6c 0a 00 08 09 0a 0b 0c 0d 0e

0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31

32 33 34 35 36 37

20) 01/23/2013:18:48:58 GMT Informational 0-PPE-0 : ACL ACL_PKT_LOG 213 0 : Source 10.102.56.99

--> Destination 10.102.56.45 - Protocol ICMP -Type 8 - Code 0 -TimeStamp 92617209(ms) - Hitcount 6 -

Hit Rule ACL2 - Action DENY - Data 08 00 c6 a6 7e 73 00 61 1e c4 ff 50 9f 6c 0a 00 08 09 0a 0b 0c 0d 0e

0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31

32 33 34 35 36 37

[From Build 112.15] [#290631]

- Configure Stateless NAT46 Translation

  The stateless NAT46 feature enables the communication between IPv4 and IPv6 networks, by way of IPv4 to IPv6 packet translation and vice versa, without maintaining any session information on the NetScaler appliance.

  A stateless NAT46 configuration on the NetScaler appliance has the following components:

  * IPv4-IPv6 INAT entry. An entry defining a 1:1 relationship between a public IPv4 address and an IPv6 address. In other words, a public IPv4 address on the appliance listens to connection requests on behalf of an IPv6 server.

  * NAT46 IPv6 prefix. A global IPv6 prefix of length 96 bits (128-32=96) configured on the appliance. During IPv4 packet to IPv6 packet translation, the appliance sets the source IP address of the translated IPv6 packet to a concatenation of the NAT46 IPv6 prefix [96 bits] and the IPv4 source address [32 bits] that was received in the request packet.

  [From Build 112.15] [#284926]

- IPv6 Protocol Compliance

  You can configure multiple link-local addresses as type SNIP6. A link-local SNIP6 address can be bound to only one VLAN, and a VLAN can have only one link-local SNIP6 address. Because NetScaler owned IP addresses are of type floating, the link-local SNIP6 address bound to a VLAN is associated with all the interfaces bound to the VLAN. Any Neighbor Discovery for IPv6 (ND6) traffic going out of the interface is sourced as the link-local address associated with the interface, as specified by RFC 4861.

  [From Build 112.15] [#286577]

- Stateful NAT64 Translation

  The stateful NAT64 feature enables communication between IPv4 clients and IPv6 servers through IPv6 to IPv4 packet translation, and vice versa, while maintaining session information on the NetScaler appliance.

  A stateful NAT64 configuration on the NetScaler appliance has the following components:

  * NAT64 rule: An entry consisting of an ACL6 rule and a netprofile, which consists of a pool of NetScaler owned SNIPs.

  * NAT64 IPv6 Prefix: A global IPv6 prefix of length 96 bits (128-32=96) configured on the appliance.

  When an IPv6 request packet received by the NetScaler appliance matches an ACL6 defined in a NAT64 rule and the destination IP of the packet matches the NAT64 IPv6 prefix, the NetScaler appliance considers the IPv6 packet for translation.

  The appliance translates this IPv6 packet to an IPv4 packet with a source IP address matching one of the IP address in the netprofile defined in the NAT64 rule, and a destination IP address consisting of the last 32 bits of the destination IPv6 address of the IPv6 request packet. The NetScaler appliance creates a session and forwards the packet to the IPv4 server. Subsequent responses from the IPv4 server and requests from the IPv6 client are translated accordingly by the appliance for the duration of the session.

  [From Build 112.15] [#316933]

- Set Interval for Generating ACL Log Messages

  You can now set the interval at which a log message is to be generated by the NetScaler appliance for a particular flow that matches an extended ACL configured on the appliance.

  To set the interval, set the AclLogTime (ACL Log Time) parameter in one of the following ways:

  *On the NetScaler command line, by running the set L3 param command.

  *In the configuration utility, by using the Configure Layer 3 Parameters dialog box (Network > Settings > Configure Layer 3 Parameters).

  [From Build 112.15] [#301716]

- The "show lacp" command does not display the lacp configurations. This issue is observed only in a cluster setup.

  [From Build 112.15] [#288450, 290635, 324248]

- Configuring Link Redundancy by using LACP channels

  Link Redundancy by using LACP channels enables the NetScaler appliance to logically create sub channels from a LACP channel where one of the sub channel is active and the remaining sub channels stay in standby mode. If the active sub channel fails or does not meet a minimum threshold throughput, one of the standby sub channel takes over and becomes active.

  The NetScaler appliance forms a sub channels from links that are part of the LACP channel and are connected to a particular device. For example, for a LACP channel with four interfaces on a NetScaler appliance, where two of the interface is connected to device A, and the other two interfaces are connected to device B, then the NetScaler appliance logically creates two sub channels, one sub channel with two links to device A, and the other sub channel with

the remaining two links to device B.

The lrMinThroughput parameter is introduced for configuring link redundancy for a LACP channel. This parameter specifies the minimum throughput threshold to be met by the active sub channel of a LACP channel. When the throughput of the active channel falls below the lrMinThroughput , link failover occurs and one of the standby sub channels becomes active.

For example, set channel la/1 -lrMinThroughput 2000

Link redundancy for a LACP channel is disabled, which is also the default setting, when you set the lrMinThroughput parameter of the LACP channel to zero or when you unset this parameter.

Note: In an HA configuration, if you want to configure throughput (throughput parameter) based HA failover and link redundancy (lrMinThroughput parameter) on a LACP channel, you must set a lesser or equal value to the throughput parameter as compared to the lrMinThroughput parameter.

For example, set channel la/1 throughput 2000 -lrMinThroughput 2000

HA failover does not occur if any of the sub channels meets the lrMinThroughput parameter value even when the total throughput of the LACP channel does not meet the throughput parameter value.

HA failover occurs only when the entire sub channels of the LACP channel does not meet the lrMinThroughput parameter value and the total throughput of the LACP channel does not meet the throughput parameter value.

For more information, see http://edocssand.citrix.com/proddocs/topic/ns-system-10-5-map/ns-nw-config-lr-lacp-tsk.html.

[From Build 120.13] [#346763]

- Now, the NetScaler appliance sends all ARP replies from the first interface (lexicographical order) of an LA channel.

  [From Build 129.22] [#486632]

- The MPX 22040/22060/22080/22100/22120 platform now supports NetScaler release 9.3 build 65.x.

  [From Build 121.10] [#311561]

- NetScaler MPX appliances now support Cisco QSFP+ cables (part number L45593-D178-C30).

  [From Build 122.17] [#427155]

- Increased Throughput on the NetScaler MPX 5650 Appliance

  The MPX 5650 appliance now delivers a throughput of 5Gbps.

  [From Build 123.11] [#428131, 432315]

- New NetScaler MPX Appliances

  Release 10.1-122.x supports the new MPX 8005 and MPX 8015 appliances.

  [From Build 123.11] [#421834, 428128]

- If an LCD hardware failure is detected on a NetScaler MPX appliance, the appliance restarts. With this enhancement, the LCD application gracefully exits without restarting the appliance.

  [From Build 123.11] [#430690]

- New NetScaler SDX Appliance

  Release 10.1-122.x supports the new SDX 8015 appliance.

  [From Build 123.11] [#429429, 432915]

- Support for ECDHE Ciphers

  The Citrix NetScaler MPX 11515/11520/11530/11540/11542 appliances support the ECDHE cipher group. On the SDX 11515/11520/11530/11540/11542 appliances, the cipher group is supported only if an SSL chip is assigned to a VPX instance. This group contains the following ciphers:

  - TLS1-ECDHE-RSA-RC4-SHA

  - TLS1-ECDHE-RSA-DES-CBC3-SHA

  - TLS1-ECDHE-RSA-AES128-SHA

  - TLS1-ECDHE-RSA-AES256-SHA

  The following ECC curves are supported:

  - P_256

  - P_384

  - P_224

  - P_521

  Note: ECC curves 224 and 521 are not supported with TLS1.2 protocol.

  [From Build 124.13] [#453765]

- The 10G ixgbe (ix) driver on the NetScaler appliance now supports the following Active Optical Cables (AOCs):

  - Finisar FCBG110SD1C03

  - Avago AFBR-7CAR03Z

  [From Build 125.9] [#419237]

- The SDX 24100/24150 and MPX 24100/24150 platforms are now supported in this release.

  [From Build 129.22] [#487831]

- The MPX 25100T and MPX 25160T platforms are now supported in this release. For more information about these platforms, see http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-platforms-con/ns-hardware-25100T-25160T-ref.html.

[From Build 132.8] [#486703, 495591, 552218]

- LDAP Referral Support

  AAA now supports LDAP referrals. If this feature is enabled, and the NetScaler appliance receives an LDAP_REFERRAL response to a request, AAA follows the referral to the active directory (AD) server contained in the referral and performs the update on that server, First, AAA looks up the referral server in DNS, and connects to that server. If the referral policy requires SSL/TLS, it connects via SSL/TLS. It then binds to the new server with the binddn credentials that it used with the previous server, and performs the operation which generated the referral. This feature is transparent to the user.

  LDAP referral support is disabled by default, and must be explicitly enabled for each ldapAction. This feature cannot be turned on globally. The system administrator must also make sure that the AD server accepts the same binddn credentials that are used with the referring (GC) server.

  To enable LDAP referrals, type the following commands at the NetScaler command line:

  set authentication ldapAction <name> -followReferrals ON

  set authentication ldapAction <name> -maxLDAPReferrals <integer>

  For <integer>, substitute the maximum level of referrals. By default, one referral level is allowed.

  For more information, see http://support.citrix.com/proddocs/topic/ns-security-10-1-map/ns-aaa-setup-policies-auth-ldap-tsk.html.

  [From Build 112.15] [#327591]

- HTTP Callouts

  HTTP callout responses can now be cached for a specified time duration.

  [From Build 112.15] [#233253, 241059]

- HTTP Callouts

  HTTP callouts now support IPv6 addresses.

  [From Build 112.15] [#215794]

- HTTP callouts can now generate HTTPS requests. When configuring the HTTP callout, you must set the "scheme" parameter of the "set policy httpCallout" command.

  [From Build 112.15] [#317392]

- Support for Hashing Text Strings

  You can now hash text strings by using the following algorithms: MD2, MD4, MD5, SHA1, SHA224, SHA256, SHA384, and SHA512. The method provided for this purpose is DIGEST(algorithm) and it can be used on text strings. For example, to hash the body of a HTTP request by using MD5 algorithm, the expression is: HTTP.REQ.BODY(1000).DIGEST(MD5)

  [From Build 112.15] [#236496, 246627]

- TCP Level Expressions

  You can now get the smoothed round trip time and the bandwidth of TCP connections for the client and server by using the following expressions:

  * CLIENT.TCP.SMOOTHRTT

  * CLIENT.TCP.BANDWIDTH

  * SERVER.TCP. SMOOTHRTT

  * SERVER.TCP.BANDWIDTH

  [From Build 112.15] [#236816]

- You can now specify an expression that produces a body of the HTTP callout. The expression must be specified in the -bodyExpr parameter of the "set httpCallout" command. A "Content-Length" header is automatically added with an appropriate value indicating that the request message contains a body. You can use the "unset httpCallout" command with the -bodyExpr parameter when you do not want to use the body expression for the HTTP callout.

  [From Build 112.15] [#340586]

- Get Information of RPC Request

  You can now get information about an RPC request by using the following expressions:

  * MSSQL.REQ.RPC.BODY?Returns the body of the SQL request as a string in the form of parameters represented as "a=b" clauses separated by commas, where "a" is the RPC parameter name and "b" is its value.

  * MSSQL.REQ.RPC.BODY(n)?Returns part of the body of the SQL request as a string in the form of parameters represented as "a=b" clauses separated by commas, where "a" is the RPC parameter name and "b" is its value. Parameters are returned from only the first "n" bytes of the request, skipping the SQL header. Only complete name-value pairs are returned.

  Both expressions return text data, on which any text operation can be performed.

  [From Build 112.15] [#320216]

- You can now get the ethertype by using an advanced policy expression.

  Examples:

  - CLIENT.ETHER.ETHERTYPE.EQ(IPv4)

  - SERVER.ETHER.ETHERTYPE.EQ(IPv6)

  [From Build 129.22] [#388879]

- The following SNMP counters for IPv6 are added in snmp group nsIp6StatsGroup:

  - ipv6TotRxPkts : IPv6 packets received

  - ipv6TotTxPkts:: IPv6 packets transmitted

- ipv6TotRxBytes: IPv6 bytes received

- ipv6TotTxBytes: IPv6 bytes transmitted

- ipv6FragTotRxPkts:IPv6 Fragments received

- ipv6FragRxPkts: IPv6 Fragments received

- ipv6FragTotPktsForward : IPv6 Fragments bridged

- ipv6FragTotPktsProcessNoReass : IPv6 Fragments processed without reassembly

- ipv6ErrHdr : IPv6 error hdr packets

- ipv6LandAttack: Land-attack packets received

- ipv6FragZeroLenPkt : Packets received with a fragment length of 0 bytes

- ipv6TotIcmpFragPkts : ICMPV6 fragmented packets

- ipv6TotLookupDone : IPV6 Neighbour Look ups.

- ipv6TotLookupFailed: IPV6 Neighbour Look ups failed

- ipv6TotStaticRoutes : IPV6 Static Routes

- ipv6TotDynamicRoutes : IPV6 Static Routes.

- ipv6TotNeigborDiscovered : IPV6 Total Neighbor Discovered.

- ipv6TotIpv6To4Conversions: IPV6 To IPV4 Conversions.

- ipv6TotIpv4To6Conversions : IPV4 To IPV6 Conversions.

- ipv6TotTcpConnection : IPV6 TCP Connections.

- ipv6TotNonTcpConnection : IPV6 Non TCP Connections.

[From Build 112.15] [#339095]

- The owner node for the SNMP engine can be set in a cluster. Use the ownerNode parameter of the set SNMP engineID command.

  [From Build 112.15] [#356223]

- A new SNMP OID, vsvrEstablishedConn (1.3.6.1.4.1.5951.4.1.3.1.1.71) is available for current client connections in the ESTABLISHED state at the vserver level.

  [From Build 126.12] [#418044]

- Support for TLS1.1 and TLS 1.2

  The SSL virtual server on the NetScaler appliance supports TLS1.1 and TLS1.2 protocol based clients. These protocols helps prevent Browser Exploit Against SSL/TLS (BEAST) attacks.

For more information about this protocol, see https://tools.ietf.org/html/rfc5246.

The following ciphers support the TLS1.1 and TLS1.2 protocol:

- SSL3-RC4-MD5

- SSL3-RC4-SHA

- SSL3-DES-CBC3-SHA

- TLS1-AES-256-CBC-SHA

- TLS1-AES-128-CBC-SHA

- SSL3-EDH-RSA-DES-CBC3-SHA

- TLS1-DHE-RSA-AES-256-CBC-SHA

- TLS1-DHE-RSA-AES-128-CBC-SHA

The following ciphers support the TLSv1.1 protocol:

- SSL3-DES-CBC-SHA

- SSL3-EDH-RSA-DES-CBC-SHA

- SSL3-ADH-RC4-MD5

- SSL3-ADH-DES-CBC-SHA

- SSL3-ADH-DES-CBC3-SHA

- TLS1-ADH-AES-128-CBC-SHA

- TLS1-ADH-AES-256-CBC-SHA

[From Build 112.15] [#271648, 205184, 258052, 258328, 262506, 315852]

- Configuring SSL Close-notify at the Entity Level

  Although the global sendCloseNotify parameter must be set to YES if any entity is to send an SSL close-notify, an entity no longer has to inherit this setting from the global settings. You can set the sendCloseNotify parameter at the entity (virtual server, service, or service group) level. This enhancement provides the flexibility to set this parameter for one entity and unset it for another entity. However, make sure that you set this parameter at the global level. Otherwise, the setting at the entity level does not apply.

  [From Build 112.15] [#257122]

- Support for SPDY in SSL

  The NPN extension is now supported on the NetScaler appliance.

  [From Build 112.15] [#284270, 329666, 329672]

- Add a Certificate Bundle

You can load a certificate bundle containing one server certificate, up to nine intermediate certificates, and optionally, a server key. Separate steps for loading and linking the certificates are no longer required.

[From Build 112.15] [#236585, 277630]

- Certificate Expiry Monitoring

  The certificate expiry monitoring option is now enabled by default, and the default expiry notification period is set to 30 days.

  [From Build 112.15] [#351522]

- Restrict the Root CA's distinguished names (DN) sent by the NetScaler Appliance

  As a part of the SSL handshake, in the Certificate Request message during client authentication, the server lists the distinguished names (DNs) of all the certificate authorities (CAs) bound to the server from which it will accept a client certificate. If you do not want the DN name of a specific CA certificate to be sent to the SSL client, set the skipCA flag. This setting indicates that the particular CA certificate's distinguished name should not be sent to the SSL client.

  [From Build 112.15] [#262041]

- Low Encryption Licenses for Russia

  A NetScaler MPX appliance for customers in Russia initially ships with a low encryption license. After proper authorization from the Russian agency, customers can upgrade to a Standard, Enterprise, or Platinum software edition, which enables high-encryption SSL performance on the appliance.

  [From Build 118.7] [#349674, 379439]

- As part of the SSL handshake with the server, the NetScaler appliance now sends a Client Hello message on the basis of the version (for example SSLv3 or TLS1.0) that is configured on the appliance. Earlier, it sent an SSLv2 compliant Client Hello message to the server.

  [From Build 123.11] [#378806, 204465, 406907]

- Setting the Limit for Disabled SSL Chips

  You can now set a limit to the number of disabled SSL chips after which the appliance restarts. At the command prompt, type:

  set ssl parameter ?cryptodevDisableLimit <positive_integer>

  A chip is marked disabled after the third failed reinitialization attempt.

  [From Build 125.9] [#376153]

- An SSL chip is disabled at the third reinitialization attempt. That is, the maximum reinitialization limit is 2. Earlier, this limit was 5.

  [From Build 125.9] [#455821]

- Display HSM Model Number

  The output of the "show fips" command now displays the HSM model number as shown below. This is especially helpful

if you are conducting an audit of the FIPS card in a NetScaler appliance and cannot open the appliance without voiding the warranty.

> sh fips

FIPS HSM Info:

HSM Label : NetScaler FIPS

Initialization : FIPS-140-2 Level-2

HSM Serial Number : 2.1G1037-IC000253

HSM State : 2

HSM Model : NITROX XL CN1620-NFBE

Hardware Version : 2.0-G

Firmware Version : 1.1

Firmware Release Date : Jun04,2010

Max FIPS Key Memory : 3996

Free FIPS Key Memory : 3994

Total SRAM Memory : 467348

Free SRAM Memory : 62580

Total Crypto Cores : 3

Enabled Crypto Cores : 3

Done

[From Build 129.22] [#385499]

- On all the NetScaler MPX platforms, DH cryptographic operation is now offloaded to the hardware, reducing the load on the CPU. If your deployment uses DH crypto operations heavily, you will notice a performance improvement.

  [From Build 131.11] [#490273, 378182, 404081]

- Clearing Statistical Counters

  You can now clear the counters that are displayed by the configuration utility's Dashboard and by stat commands in the NetScaler command-line interface. Clearing a counter resets it to zero, from which point it is incremented as the appliance processes traffic. You can clear the counters regardless of whether the NetScaler appliance is currently processing traffic. The ability to clear counters enables you to observe them over a specific period of time and troubleshoot the configuration.

  [From Build 112.15] [#241836, 189957, 192898, 228298]

- The Citrix NetScaler MPX 22040/22060/22080/22100/22120 appliances now support the ECDHE cipher group. This group contains the following ciphers:

  - TLS1-ECDHE-RSA-RC4-SHA

  - TLS1-ECDHE-RSA-DES-CBC3-SHA

  - TLS1-ECDHE-RSA-AES128-SHA

  - TLS1-ECDHE-RSA-AES256-SHA

  Because of its smaller key size, Elliptic Curve Cryptography (ECC) is especially useful in a mobile (wireless) environment and in an interactive voice response environment, where every millisecond is important. Smaller key sizes result in power, memory, bandwidth, and computational cost savings.

  The following ECC curves are supported:

  - P_256

  - P_384

  - P_224

  - P_521

  By default all four curves are bound to an SSL virtual server.

  For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-5-map/ns-ssl-config-ecdhe-ciphers-tsk.html

  [From Build 121.10] [#329257, 198673, 401256]

- Enabling CallHome Feature while Upgrading the NetScaler Appliance

  While upgrading the NetScaler appliance from an older release to release 10.1 or later, the NetScaler appliance prompts you to enable the CallHome feature in one of the following cases:

  - The CallHome feature is not supported in the older release.

  - The CallHome feature is disabled in the older release.

  [From Build 112.15] [#311617]

- Send Buffer Support for TCP Profiles

  You can now set the window that is advertised to the server by using the sendBuffsize parameter of the "set ns tcpProfile" command.

  [From Build 112.15] [#315625]

- User Name and Password Length Extended to 127 Characters

User names and passwords on the NetScaler appliance can now be up to 127 characters in length. Usernames and passwords can consist of upper-case and lower-case letters, digits, and the hyphen and underscore characters.

[From Build 112.15] [#325421]

- Public Key Authentication for Non-nsroot Users

  All NetScaler users can now access the NetScaler appliance by using public key authentication in SSH.

  [From Build 112.15] [#209190, 235961, 291483]

- New Parameters for Web Interface Site

  The following parameters are added for a web interface site:

  For the add wi site command:

  - welcomeMessage. Localized welcome message that appears on the welcome area of the login screen.

  - footerText. Localized text that appears in the footer area of all pages.

  - loginSysMessage. Localized text that appears at the bottom of the main content area of the login screen.

  - appWelcomeMessage. Localized text that appears at the top of the main content area of the applications screen.

  - preLoginButton. Localized text that appears as the name of the pre-login message confirmation button.

  - preLoginMessage. Localized text that appears on the pre-login message page.

  - preLoginTitle. Localized text that appears as the title of the pre-login message page.

  - showSearch. Enables the Search option on XenApp websites.

  - showRefresh. Provides the Refresh button on the applications screen.

  - wiUserInterfaceModes. Appearance of the login screen.

  ---- Simple - Only the login fields for the selected authentication method are displayed.

  ---- Advanced - Displays the navigation bar, which provides access to the prelogin messages and preferences screens.

  - userInterfaceLayouts. Specifies whether or not to use the compact user interface.

  - domainSelection. Domain names listed on the login screen for explicit authentication.

  For the bind wi site command:

  - farmName. Name for the logical representation of a XenApp or XenDesktop farm to be bound to the Web Interface site.

  - groups. Active Directory groups that are permitted to enumerate resources from server farms. Including a setting for this parameter activates the user roaming feature. A maximum of 512 user groups can be specified for each farm defined with the Farm<n> parameter. The groups must be comma separated.

  - recoveryFarm. Binded farm is set as a recovery farm.

[From Build 112.15] [#317793]

- SNMP statistics can be cleared by using the clearstats parameter of the stat snmp command.

  [From Build 112.15] [#362132]

- SPDY Support

  NetScaler appliances can now support SPDY. You have to enable SPDY in an HTTP profile and bind the profile to a virtual server. When SPDY is enabled, the virtual server functions as a SPDY gateway and converts SPDY requests from the clients into HTTP requests that it sends to the servers. It also converts the HTTP responses from the servers to SDPY responses that it sends to the clients. The servers do not have to support SPDY. You can enable SPDY in an HTTP profile by using the set ns httpprofile - SPDY enabled command or by using the configuration utility.

  Note: SSL is required for SPDY protocol to function.

  [From Build 112.15] [#329671]

- PHP Version Upgraded from 5.3.10 to 5.3.17

  The PHP version has been upgraded from 5.3.10 to 5.3.17 on the NetScaler appliance to resolve security vulnerabilities and stability issues with PHP.

  [From Build 112.15] [#333572]

- Multipath TCP Support

  NetScaler appliances now support Multipath TCP (MPTCP). MPTCP is a TCP/IP protocol extension that identifies and uses multiple paths available between hosts to maintain the TCP session. You have to enable MPTCP on a TCP profile and bind it to a virtual server. When MPTCP is enabled, the virtual server functions as an MPTCP gateway and converts MPTCP connections with the clients to TCP connections that it maintains with the servers.

  For more information, see TCP Configurations.

  [From Build 119.7] [#320221, 307024]

- Custom HTTP Headers Support using Web Server Logging

  The NetScaler can now export values of custom HTTP headers to the NSWL client. You can configure up to a maximum of two HTTP request header names and two HTTP response header names.

  [From Build 119.7] [#329710]

- Call Home Proxy Mode Support

  Call Home can now upload your NetScaler appliance's data to the Citrix TaaS server through a proxy server.

  [From Build 119.7] [#311623]

- Backing Up and Restoring a NetScaler Appliance

  You can now back up the NetScaler appliance at any time and then use the backup to restore the same appliance to that state.

For more information, see Backing Up and Restoring the NetScaler Appliance.

[From Build 119.7] [#367021]

- New Subnet Mask Field for the SNIP Address in the First-time Setup Wizard

  The first-time setup wizard now has separate subnet mask fields for the NetScaler IP (NSIP) and subnet IP (SNIP) addresses.

  [From Build 120.13] [#413542]

- Before reusing a server connection in the reuse pool, the NetScaler appliance checks the connection's idletimeout and reusepool values, and closes the connection if either value is exceeded. The appliance also checks the reuse pool for idle connections, and closes them, more frequently than specified by the zombie timer interval.

  [From Build 122.17] [#365828, 365731]

Fixed Issues in Previous 10.1 Builds

Oct 10, 2015

The issues that were addressed in NetScaler 10.1 releases prior to Build 133.9. The build number provided below the issue description indicates the build in which this issue was addressed.

AAA-TM | AAA-TM/Content Switching | Action Analytics | AppFlow | Application Firewall | Cache Redirection | Citrix NetScaler 1000V | CloudBridge Connector | Cluster | Command Line Interface | Compression | Configuration Utility | Configuration utility | Content Switching | DNS | DataStream | Documentation | GSLB | Graphical User Interface | High Availability | ICA AppFlow | Integrated Caching | Load Balancing | Load Balancing/AAA-TM | Load Balancing/DNS | Load Balancing/MSSQL | Load Balancing/Responder | MPTCP | Monitoring | NITRO API | NetScaler Gateway | NetScaler Insight Center | NetScaler SDX Appliance | NetScaler VPX Appliance | Networking | Platform | Policies | Policies [#446507,#506761,#463284,#500444] | Policy | Rewrite | SNMP | SPDY | SSL | SureConnect | System | User Interface | WIonNS | Web Interface | XML

- During authentication, when AAA generated a URL redirect, it rewrote the query portions of URLs that contained Base64 strings into base 8 ASCII string equivalents instead of transmitting the original strings. This caused some redirects to fail, and introduced security issues into other redirects. This behavior has been changed, and AAA now transmits the unmodified query to the user. Users should be aware that the new approach might cause issues with different protected web applications.

  [From Build 118.7] [#390037, 242875, 246109, 247244, 358370]

- When Kerberos Constrained Delegation is configured with a content switching virtual server, the NetScaler appliance might hang or crash. The cause is a GET request with multiple authorization headers. (Only one authorization header is expected.)

  [From Build 118.7] [#372362, 381621, 401539]

- On a NetScaler appliance with AAA enabled and Kerberos Constrained Delegation single sign-on configured, after several single sign-on requests are successfully authenticated, the virtual server principle can unexpectedly become blank. When this happens, subsequent authentication requests fail.

  [From Build 118.7] [#387076, 390083]

- When importing a keytab while setting up a KCD account, AAA might fail to extract the SPN from the keytab, causing the import to fail.

  [From Build 119.7] [#387049]

- When AAA is configured by authentication profile on a NetScaler appliance that has content switching enabled, users can use the Microsoft Internet Explorer or Mozilla Firefox browsers to log on, but might not be permitted to access all resources that they should be able to access. Users who log on using the Google Chrome browser do not experience this problem. The underlying cause was that authentication level is checked only once per connection rather than at each request.

  [From Build 120.13] [#401000]

- On a NetScaler ADC that has AAA-TM enabled and Kerberos authentication configured, when you direct traffic through the ADC to a Microsoft SQL server, an error causes the ADC to restart.

  [From Build 123.11] [#436493]

- When AAA-TM is configured to use SAML authentication, the redirect URL that the SAML virtual server returns appends the string "%00", a text-based form of the null value, to the original redirect URL. Most browsers handle the appended string properly, but newer Apple iOS and some Apple MacOS browsers fail to load the web page because of this string.

  To work around this issue, you can create a Rewrite action and policy to strip off the "%00" string, and bind it to global. If you configure the gotoPriorityExpr for the policy to NEXT, and bind the policy with a priority of 1, it will run first, strip the null string from the end of all redirect URLs, and then continue policy evaluation with the next policy. This configuration should work without creating any problems with your existing policy evaluation flow.

  [From Build 124.13] [#441755]

- RFC822 Name-based Certificate Authentication

  AAA-TM now supports the use of RFC822 name-based (SAN) client certificates to authenticate users. SAN client certificates work in exactly the same way as other client certificates. To configure the NetScaler ADC to use SAN client certificate authentication, follow the client certificate authentication instructions in the AAA-TM documentation.

  [From Build 125.9] [#453125]

- When the NetScaler ADC is configured to use AAA with SAML authentication, and it receives a response from the IDP, it reformats the response in standard SAML format. (This process is sometimes called "canonicalizing" the response.) The ADC might not reformat SAML <samlp: response> namespace prefix tags correctly, because it expects <saml: assertion> format. In that case, digest verification fails.

  To work around this issue, you must remove the namespace prefixes definition, as described on the following web page:

  http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=%2Fcom.ibm.tivoli.fim.doc_6226%2Fconfig%2Freference%2FCustomPropsKess.html

  [From Build 125.9] [#435529, 448986, 456574]

- AAA-TM now supports relative URLs as form Action URLs in forms-based SSO logon forms. You do not have to specify an absolute path to the web form when configuring forms-based SSO.

  [From Build 127.10] [#317157]

- When AAA is configured to authenticate users to a Microsoft Sharepoint 2013 server by using NTLM, the user might be prompted to retype his or her credentials even though the user entered those credentials correctly. After the user retypes the credentials, he or she is logged on successfully. The issue is that initially the NetScaler ADC sends an incorrect domain to Sharepoint.

  [From Build 129.22] [#476885]

- If the hostname that sends an incoming request does not match the domain configured on the authentication virtual server, the NetScaler ADC returns an HTTP 500 error. As a workaround, configure an authentication profile and include the hostname.

[From Build 129.22] [#488015]

- In forms-based single sign-on (SSO), if the designated response size is 0, the NetScaler ADC does not search for the complete response, as it normally would for responses with sizes above 0. It therefore fails to find the login form, and forms-based SSO authentication fails.

  [From Build 129.22] [#493308]

- The Authorization header received from the client with the user credentials for 401 based authentication for KCD was intentionally corrupted by the NetScaler ADC as "Ahoutrization" before forwarding it to the backend. To avoid the risk of decoding the user-supplied credentials by using simple base64decode, the ADC now removes the incoming authorization header containing user credentials, and inserts a new Authorization header with a Kerberos token before sending the payload to the backend application.

  [From Build 129.22] [#478374]

- The NetScaler ADC no longer sets the NSC_TMAA session cookie during a secure load balancing virtual server session.

  [From Build 129.22] [#474918, 502915]

- If a user name or password consists of UTF8 characters, basic authentication fails on the NetScaler ADC. With this fix, the ADC now passes the encoding type in the 401 challenge so that the incoming data is accurately encoded.

  [From Build 130.13] [#507386]

- The NetScaler ADC does not handle an authentication request if the incoming base64 decoded kerberos ticket is more than 10 kilobytes. This fix increases the buffer-size limit to accommodate tickets of up to 65 kilobytes.

  [From Build 130.13] [#505809, 507692]

- In a AAA-TM setup that has 401 authentication enabled on the load balancing virtual server, the NetScaler appliance can, in some cases, go down if it receives a malformed authorization header.

  [From Build 131.11] [#530792]

- The NetScaler appliance can fail if the logout of the AAA-TM session is initiated through a traffic policy. The configuration that can lead to this is of the form:

  > add tm trafficAction testAction1 -InitiateLogout ON

  > add tm trafficPolicy testPolicy1 <rule> testAction1

  [From Build 131.11] [#527651]

- The NetScaler appliance sometimes sends a 401 error message to a client that sent a valid authorization header.

  [From Build 132.8] [#532675]

- When you configure a content switching rule that is evaluated before the user authenticates with AAA-TM, and the rule is supposed to redirect users to a specific virtual server on the basis of the user name, the rule fails.

  [From Build 118.7] [#397673]

- The NetScaler crashes due to an issue in hash calculation and comparison of the action analytics records. The crash is observed when the NetScaler receives URLs that differ only in case.

  Examples:

  http://10.217.6.239/TesT/

  http://10.217.6.239/TEST/

  http://10.217.6.239/TEsT/

  http://10.217.6.239/TeST/

  Note post fix:

  Stream analytics record creation will be case sensitive. For example, WWW.GOOGLE.COM and www.google.com will result in two seperate records.

  If this is not desired, stream selector results should be converted to one case. Example:

  add stream selector sel1 HTTP.REQ.hostname.to_lower

  [From Build 130.13] [#406457]

- If you enable AppFlow from a NetScaler Insight Center virtual appliance while traffic is flowing through a monitored NetScaler appliance, NetScaler Insight Center disables and then re-enables the AppFlow feature for every virtual server on the NetScaler appliance. Doing that while traffic is flowing through the appliance puts some pointer out of sync. As a result, the appliance does not respond properly.

  [From Build 118.7] [#388650, 393917, 396149, 398276, 409840]

- A newly added HTTP header prevents parsing of the HTTP request.

  [From Build 121.10] [#418296]

- A Nitro call used by NetScaler Insight Center to fetch the license information from a NetScaler appliance affects the performance of the appliance.

  [From Build 122.17] [#430591]

- The NetScaler fails to respond if appflow logging is disabled on a VPN virtual server when ICA traffic flows through the NetScaler.

  [From Build 123.11] [#430960, 470262]

- If HTML Injection is enabled, the NetScaler ADC injects JavaScript into the response to obtain client-side page-load time and client-side page-render time details. The JavaScript triggers a special request that is intended only for the NetScaler ADC, but the NetScaler ADC creates an additional request by forwarding the request to the server.

  [From Build 126.12] [#441332, 357422, 401672]

- If a browser executes the JavaScript that is inserted into the response of the main page, it sends a special request intended for the NetScaler ADC. AppFlow records for this request must not be generated. While handling this behavior, the logic in one part of the code assumes that the AppFlow records must not be sent, but another part of the code assumes that the records must be sent. As a result, the NetScaler ADC fails to respond.

  [From Build 128.8] [#478480, 480535, 495201]

- If you enable Appflow for ICA on a NetScaler ADC, the NetScaler ADC might fail under certain conditions while parsing the ICA frames.

  [From Build 129.22] [#512321, 519402]

- If you have enabled AppFlow for ICA on a NetScaler ADC, the ADC crashes while processing CGP packets.

  [From Build 129.22] [#523088]

- NetScaler ADC might fail if you disable AppFlow or clear the AppFlow actions and policies when ICA traffic flows through the NetScaler ADC.

  [From Build 129.22] [#487686, 502208, 516910]

- The NetScaler ADC fails if AppFlow is enabled and it receives an ICA command longer than 2048 bytes.

  [From Build 129.22] [#504990, 508918]

- The HTML Injection JavaScript is incorrectly inserted into one of the JavaScript responses sent by the server, causing the page to fail to load.

  [From Build 131.11] [#472971]

- AppFlow should not export the records for internal connections, like the Kernel RPC. When it attempts to export records for such an internal connection, it leads to AppFlow failure.

  [From Build 132.8] [#547892, 531101]

- The application firewall now supports sessionless cookie proxying on NetScaler cluster configurations that do not use the spotted VIP feature.

  [From Build 118.7] [#351544]

- Application Firewall Signatures

  To improve performance, when the application firewall processes buffer overflow signatures it does not evaluate PCRE expressions unless the minLength parameter is set.

  [From Build 118.7] [#376437, 365941]

- Application Firewall Signatures

  If you configure an application firewall profile but do not bind any signatures to it, the NetScaler appliance becomes unresponsive or fails if a user sends a request with a JSON body to a web site protected by that profile.

[From Build 118.7] [#390804, 393588]

- Application Firewall Signatures

  You can now configure the JSON content types for your application firewall in the "Manage JSON Content Types" dialog box in the global settings. The dialog box is nearly identical to the "Manage XML Content Types" dialog box.

  [From Build 118.7] [#384103]

- The application firewall includes an extraneous line break in the hidden field that it adds to forms as part of the form field consistency check. This line break is not javascript-compliant and can cause issues with javascript-enhanced forms.

  [From Build 119.7] [#403027]

- Application firewall statistics are not supported for NetScaler classic policies. If you need to see numbers of policy hits and other statistics, you must use NetScaler default syntax policies.

  [From Build 120.13] [#303060]

- If the NetScaler appliance sends a large amount of input data to the application firewall in a short time, the appliance can become unresponsive or fail. The appliance now sends input data in batches limited to sizes that do not cause this problem.

  [From Build 121.10] [#416714]

- On a NetScaler appliance with both the application firewall and integrated caching enabled, a memory leak might occur. To work around this issue, disable integrated caching.

  [From Build 122.17] [#391317, 423289]

- On a NetScaler appliance with the application firewall enabled, web forms submitted with URL-encoded double-byte character (Chinese, Japanese, or Korean) inputs might generate a Form Field consistency check violation. The reason is that the application firewall counts bytes instead of characters when validating web form input, causing some double-byte input to exceed the form field maxlength attribute.

  [From Build 122.17] [#422639, 239207]

- On a NetScaler appliance with the application firewall enabled and configured, if a protected web site contains a multipart web form, a memory leak causes a small amount of memory to be consumed and not released each time the application firewall processes the web form. Repeated processing of requests and responses can gradually consume available memory.

  [From Build 122.17] [#422919, 423289]

- By default, the application firewall's SQL Injection signatures patterns and security checks do not prevent SQL injection attacks that use the percent (%) or underscore (_) characters. To work around this issue, add the percent and underscore characters to each signatures object as SQL special characters.

  [From Build 123.11] [#407347]

- If memory utilization is high on a NetScaler appliance that has the application firewall enabled and configured, URL redirect might fail, causing the appliance to crash. To work around this issue, reduce memory utilization by reducing session timeouts and disabling memory-intensive filtering rules.

[From Build 123.11] [#427717]

- The application firewall currently miscalculates memory limits on 12 GB, 2 vCPU NetScaler appliances. For example, when the appliance has 2 GB of memory available, the application firewall shows only 600 MB of available memory.

  [From Build 123.11] [#427857]

- On a NetScaler MPX5500 appliance that has the application firewall enabled, and has logging enabled for at least one signature or security check, when that logging action is triggered the appliance might hang or crash.

  [From Build 123.11] [#423861, 436918]

- If you use the single sign on (SSO) feature on your NetScaler ADC or NetScaler Gateway, it might become unresponsive or restart.

  [From Build 123.11] [#446304, 443080, 444746, 444810, 447206, 448814, 449393, 449396, 451162, 451860, 452078, 452427, 453146, 454416]

- The application firewall blocks XML requests that have empty bodies (zero content length), which causes autodiscover and other features that use such requests to fail. To work around this issue, you can disable the XML Format, XML Message Validation, XML Denial of Service, and Web Services Interoperability (WSI) security checks.

  [From Build 123.11] [#432276]

- When using CVPN or the application firewall credit card or safe object security checks, memory issues might cause the NetScaler ADC to become unresponsive or restart.

  [From Build 123.11] [#448961, 449223, 449851, 450070]

- On a NetScaler ADC HA pair configured to use the Citrix VPN, single sign-on, and the Application Firewall, a memory page issue might cause the primary ADC to reboot, failing over to the secondary ADC.

  [From Build 124.13] [#445552, 367086, 444810, 450052, 453111, 453165]

- On a NetScaler appliance or VPX that has the application firewall enabled and at least one profile that has the Safe Object security check enabled, the application firewall might generate an extremely large buffer file while checking responses for objects. The oversized buffer might cause performance problems or, in extreme cases, hang the system. To work around this issue, disable the Safe Object check.

  [From Build 124.13] [#444471]

- Apple iPhone and iPad users are unable to watch MP4 videos on web sites that are protected by the application firewall when either the form field consistency check or the credit card check is enabled, even if blocking is not enabled. The problem is specific to Apple iOS. Google Android smartphone or tablet users are able to watch MP4 content.

  To work around this issue, add the following expression to the policy that invokes the application firewall:

  "HTTP.REQ.URL.REGEX_MATCH(re#.mp4$#).NOT"

  For example, to exempt URLs that contain the string ".mp4" from the policy pol_media.example.com, which calls the profile prfl_media.example.com, you would type the following command:

  > add appfw policy pol_media-example.com "HTTP.REQ.URL.REGEX_MATCH(re#.mp4$#).NOT" prfl_media.example.com

[From Build 124.13] [#405434, 412329]

- Memory Caching Issue

  On a NetScaler ADC that has the application firewall enabled, and that has either limited available memory or a small memory cache configured, a memory page issue might cause the ADC to become unresponsive or reboot.

  [From Build 125.9] [#453111]

- Viewing Large PDF Files in Google Chrome Browser

  On a NetScaler ADC that has the application firewall enabled, when a Google Chrome user opens a large PDF file on a protected web server, the ADC might become unresponsive. The same file, if downloaded with Internet Explorer or Mozilla Firefox, causes no problems. The cause is a loop in a backup queue.

  [From Build 125.9] [#452846, 438094, 453768, 456263, 459327, 461608, 464502]

- High Level of Out of Memory Errors

  On a NetScaler ADC with limited CPU and memory, if the application firewall is enabled, out-of-memory errors might accumulate in the NetScaler log, causing rapid rotation of logfiles. To work around this issue, lower the session timeout from the default 900 seconds to 360 seconds or lower.

  [From Build 125.9] [#428852]

- Response-side Check Issue with Lotus Notes Webmail

  On a NetScaler ADC that has the application firewall enabled and an XML or Web 2.0 profile configured, if a response-side check (such as the Credit Card or Safe Object check) is enabled along with at least one XML-based check, Lotus Notes webmail does not load correctly. Specifically, the frame that should contain the user's inbox is blank.

  [From Build 125.9] [#448610]

- Web Form Processing Issue Causes ADC to Become Unresponsive

  On a NetScaler ADC that has the application firewall enabled and the Form Field Consistency check or Field Formats check enabled, a memory leak might cause the ADC to become unresponsive, requiring a manual restart. The underlying issue is a failure to process certain types of web form content properly. Appliances or VPX instances that have limited CPU and memory are especially likely to experience this issue.

  To work around this issue, disable entity decoding. You can disable this feature by logging onto the NetScaler command line and, at the prompt, typing the following command:

  set appfw settings -entityDecoding off

  [From Build 125.9] [#436100]

- Web-Based Content Not Loaded Correctly when XML Checks are Enabled

  On a NetScaler ADC that has the application firewall enabled and an XML or Web 2.0 profile configured, if any XML security checks are enabled, certain web content does not load correctly. To work around this issue, create a separate profile that has the application firewall disabled. Then, create an application firewall policy that assigns that profile to those web pages that are affected by this issue.

[From Build 125.9] [#450939]

- A user with a web proxy that allows the user to modify the HTTP header can on rare occasions bypass certain security checks when sending content that would normally be blocked. For example, a user might bypass the HTML and XML SQL injection checks when sending an SQL special symbol to a protected web application, as long as the special symbol is not combined with an SQL command. A user might also be able to send a modified cookie by intercepting and including all cookies that the application firewall sent to the user, including the NetScaler cookie. Finally, the user might be able to use a web form to upload a script and save that script as a different file type.

  It does not appear that this technique can be used to cause an actual security breach.

  [From Build 126.12] [#424879]

- After automatic update of the application firewall signature rules, custom signature rules with versions lower than the current signatures are automatically disabled.

  [From Build 126.12] [#457454]

- If an attacker includes an SQL special character that is not followed by an SQL keyword in web form data filtered by the application firewall, the application firewall does not block the request because it classifies a special character that does not include a keyword as a false positive.

  [From Build 126.12] [#443207, 355620]

- Any application firewall profile that has either the "AlwaysExceptFirstRequest" or the "AlwaysExceptStartURLs" option enabled cannot be viewed in the configuration utility. These options are available from the command line only. When upgrading to either the current 10.1 maintenance release or the 10.5 beta release of the NetScaler operating system from any previous release, any profile which had the "always" option enabled has that option changed to "AlwaysExceptStartURLs." Profiles that have the "if_present" or "OFF" options enabled are not affected.

  [From Build 127.10] [#472094]

- NetScaler ADCs that are configured as an HA pair with the application firewall enabled might become unresponsive or reboot when the application firewall is processing a large web form.

  [From Build 127.10] [#455284]

- A NetScaler ADC that is configured as an HA pair, and that has the application firewall feature enabled, might experience repeated failovers from the primary to the secondary node when processing HTML traffic with large tag attribute values.

  [From Build 127.10] [#456650, 313950]

- The application firewall parses multipart forms correctly according to the appropriate RFC.

  [From Build 129.22] [#479840, 472476, 482042]

- If you update default signatures on the primary NetScaler ADC in an HA pair, you cannot sync the updated signatures to the secondary ADC.

  Workaround: Export the updated signatures, and import them on the secondary ADC.

  [From Build 129.22] [#486231]

- The SQL wildcard characters (%, _, ^, []) were accidentally removed from the Citrix application firewall default signature object. This breaks the SQL wildcard functionality when the default signature file and its clones are used. This fix restores the wildcard characters in the default signature file. The application firewall detects them and flags the SQL Injection check violations.

  [From Build 129.22] [#513952]

- If you use the configuration utility to make changes to the HTML Cross-Site Scripting check, Allowed/Denied patterns, the application firewall becomes unresponsive after the first POST request it receives after you save your changes. (The Allowed/Denied patterns are accessed through the Modify Signature dialog box.) If you use the command line to make the same changes, no problems occur.

  [From Build 129.22] [#459031, 463351]

- If the application firewall receives a multipart POST request with a Content-Type header that contains a charset, it blocks that request as malformed.

  [From Build 129.22] [#464641]

- The NetScaler application firewall "Click to Rule" functionality is not working in the 51.x and the 52.x builds of release 10.5. With this fix, the user can successfully select the pertinent log message in the syslog viewer and deploy it as a relaxation rule.

  [From Build 129.22] [#503856]

- If a NetScaler ADC receives a request for an object that it cached before the application firewall configuration was modified to add any advanced security check protection, the ADC responds with HTTP Error 503 for subsequent requests to access this cached object, because the object does not contain the expected application firewall metadata. With this fix, the existing cached objects without the required metadata are considered stale and are flushed. The request is served from the origin server and the cache is updated with refreshed data.

  [From Build 130.13] [#473322, 466491]

- The NetScaler ADC might fail if a transaction is aborted before the application firewall completes processing the request.

  [From Build 130.13] [#481899]

- When a user attempts to upload a file to a server that is protected by the application firewall, the file upload fails. The underlying cause is that the application firewall included an invalid character in the MIME boundary when encoding the file.

  [From Build 130.13] [#472476, 418036]

- The application firewall PCI-DSS report does not contain information about the "SQLInjectionCheckSQLWildChars" parameter.

  [From Build 130.13] [#423150]

- If the NetScaler application firewall receives a request with percent-encoded space character, such as "login%20name" for a form field login name, the deployed learned rule containing the encoded character (%20) fails to work as relaxation rule. The security check violation is still triggered. Note that the browser converts the space to a "+" character. For such a

request, the corresponding learned rule with "login+name" for "login name" works as expected when deployed as a startURL relaxation rule.

Workaround: Edit the relaxation rule to replace "%20" with "\\s*" for requests with percent encoded space characters.

[From Build 130.13] [#315183]

- NetScaler Application Firewall Default Signature object now has rules that can be enabled to protect against Shellshock vulnerability (CVE-2014-6271, CVE-2014-7169) which could allow arbitrary code execution.

  [From Build 130.13] [#505272, 505039]

- If a response contains href links that include query parameters, the NetScaler application firewall triggers false positives for CSRF and form field consistency violations if these links are accessed. With this fix, if CSRF or Field Consistency checks are enabled, the URLs in the hrefs are added to the URL Closure table even if startURL Closure is not enabled.

  [From Build 130.13] [#488369]

- If CEF logging is turned on, only the format of application firewall log messages is expected to change, but the format of other logs is also affected, causing problem with their display. With this fix, turning on the application firewall CEF logging does not modify the format or display of other logs.

  [From Build 130.13] [#476206]

- The Application Firewall PCI-DSS report does not display signature bindings. The Profile Settings section of the report shows bound signatures as "Not Set".

  [From Build 130.13] [#443673]

- After an upgrade from a 9.3 build, the user interfaces display inaccurate information about classic policy bindings and inheritance. With this fix, both the configuration utility and the command line interface display the information accurately.

  [From Build 131.11] [#511480]

- The external syslog servers are not able to properly display the audit-log messages from the NetScaler application firewall, because the messages are longer than expected. With this fix, the messages are the correct length.

  [From Build 131.11] [#528170]

- A 64 bit memory leak in the application firewall module might lead to cache misses. The memory leak occurs when the cache is turned on and any of the advanced application firewall security checks are enabled. The application firewall memory leak is now fixed, and the fix resolves the interoperability issue with the cache module.

  [From Build 132.8] [#549466]

- During binding a signature to an application firewall profile, the NetScaler appliance might fail when it is under memory pressure.

  [From Build 132.8] [#559060]

- The Perl script that parses and merges the application firewall signatures during an update operation can cause Perl to crash on the NetScaler ADC. The crash files reduce the amount of space available on the hard drive.

[From Build 132.8] [#543372]

- The NetScaler ADC might fail if a request attempts to access uninitialized variable for an application firewall protected resource. This might be seen when the path ends with "/..".

  [From Build 132.8] [#517750, 530793]

- Enabling the NetScaler application firewall XML Format check might block the contents of a response when the user accesses an embedded link in some applications. The response might be truncated even when the XML format check is deployed in a non-block mode.

  [From Build 132.8] [#528902, 558724]

- When any form protection check is enabled and the default request content-type parameter of the application firewall profile is not configured, an incoming request without a content-type header is treated as a form, even if it is not a form. The transfer-encoding header gets deleted, and a content-length header gets added, but the request is forwarded to the server as a chunked request. The server is unable to process the chunked data and determines it to be a bad request. With this fix, the form analysis is carried out only when "multipart/form-data" or "application/x-www-form-urlencoded" content type is either specified in the request or set as the default request content type in the profile that is applied when the content-type is not specified in the request.

  [From Build 132.8] [#559348]

- The response for an XML GET request might be truncated if, in addition to any of the XML checks, the creditcard or safeobject checks are enabled for the application firewall profile.

  [From Build 132.8] [#539777]

- The NetScaler cache fails to respond to a request in which an absolute URL does not include a slash (/) after the host name.

  [From Build 119.7] [#401148, 408856, 441788]

- An invalid HTTP request received on a cache redirection virtual server configured on the NetScaler ADC is sent to the cache server. This results in errors and degraded performance.

  With the fix, invalid HTTP requests are redirected to the origin server instead of the cache server.

  [From Build 129.22] [#497866, 502366]

- When the cache redirection virtual server is configured as a forward proxy, if an ASYNC memory allocation failure happens, the NetScaler appliance might fail to respond while trying to access a page on the a server that is already configured as a service on the NetScaler.

  [From Build 130.13] [#486578, 491485, 502030, 519399]

- Applying multiple ACL rules causes excessive consumption of CPU cycles. As a result, the NetScaler ADC might become unresponsive.

  [From Build 130.13] [#502366, 505091, 514785]

- The NetScaler ADC fails if the cache redirection virtual server and the httpport parameter point to the same service. For example, the following configuration causes the ADC to fail:

  > set ns param -httpport 80

  > add cr vserver cr1 http * 80

  > set cr vserver cr1 -listenpoliciy "client.ip.src.eq(1.1.1.1)"

  Workarounds:

  Add a listen policy when you add the cache redirection virtual server. For example:

  set ns param -httpport 80

  > add cr vserver cr1 -td 0 HTTP * 80 -range 1 -cacheType TRANSPARENT -Listenpolicy "CLIENT.IP.DST.EQ(4.4.4.10)"

  OR:

  Unset the httpport parameter. For example:

  > unset ns param httpport

  > add cr vserver cr1 http * 80

  [From Build 131.11] [#509690]

- In a fully transparent CR deployment if a client sends two HTTP GET requests for the same connection, the first connection to the CACHE is closed when the second GET request is received. This happens because a specific flag is set to open new connection which forwards the second GET request to the cache. Since the first connection for the same 4 tuple is still open, NetScaler sends a reset signal.

  Fix: Do not set the flag to initiate the connection for the second GET request, since the previous connection already exists.

  [From Build 132.8] [#541395]

- NetScaler-VSB supporting 9 virtual NICs comes up with virtual NICs. This happens when there is an existing NetScaler-VSB (pre 10.5-52.x) on Nexus1110x that supports 7 virtual NICs.

  [From Build 129.22] [#499050]

- The Internet Key Exchange Daemon (IKED) might fail after the NetScaler ADC is restarted.

  [From Build 128.8] [#460193, 444265, 451886, 474654]

- Traffic latency might be greater than 100 milliseconds in a CloudBridge connector tunnel between two NetScaler appliances.

  [From Build 129.22] [#498541]

- Memory leaks might occur on NetScaler ADCs connected to a CloudBridge Connector tunnel when one of the ADCs sends monitor probes, through the tunnel, to a service that is bound to an HTTP or SSH load balancing virtual server.

  [From Build 129.22] [#512191, 513775]

- When the state of a CloudBridge connector tunnel is DOWN, there is a delay in displaying the related log messages (from the /tmp/iked.debug file) on the Create CloudBridge Connector page of the configuration utility.

  [From Build 130.13] [#440781]

- A newly added node cannot synchronize the cluster configuration, because it cannot establish a connection to the cluster configuration coordinator. This issue might arise if the configuration coordinator rpcNode password on the new node is not the same as that on the configuration coordinator.

  Workaround: Make sure the configuration coordinator rpcNode password on the newly added node is the same as that of the configuration coordinator. If not, use the "set ns rpcNode" command to update the password.

  [From Build 118.7] [#370814]

- In some cases, the MSR routes remain in DOWN state since probing ownership is incorrectly being distributed across the cluster. MSR in cluster needs spotted SNIPs and probing ownership must be with the local node alone.

  [From Build 126.12] [#455148]

- When upgrading a cluster node to NetScaler 10.5, from any build of NetScaler 10.1, make sure that the "syncookie" parameter is disabled on the TCP profiles. Otherwise, there can be disruption in traffic flow.

  [From Build 129.22] [#480071, 483171]

- From NetScaler 10.5 Build 52.x, the cluster feature is licensed with the Platinum and Enterprise licenses. In earlier releases, the cluster feature was licensed by a separate cluster license file.

  Note:

  - If you have configured a cluster in an earlier build, the cluster will work with the separate cluster license file. No changes are required.

  - When you configure a new cluster in Build 52.x and then downgrade to an earlier build, the cluster will not work as it now expects the separate cluster license file.

  [From Build 130.13] [#486259]

- NetScaler cluster nodes may send a large number of ARP requests if a large number of ARP entries are learned over a cluster LA interface.

  [From Build 132.8] [#519327, 542633]

- The "show ns runningConfig" command displays the current time instead of the time at which the configuration was last modified.

[From Build 121.10] [#379234]

- After a user logs on to a NetScaler appliance through the CLI, the "set cli mode -disabledFeatureAction NONE" command is automatically executed, and the following error message appears:

  ERROR: Not authorized to execute this command.

  [From Build 122.17] [#420596]

- A policy bound to a vpn vserver with "-type RESPONSE" gets lost after a reboot. That is, it is no longer bound after a reboot.

  [From Build 125.9] [#441505]

- When you run the command show techsupport to generate a tar of system configuration data, in certain scenarios, the NetScaler ADC might ignore to collect certain large files.

  [From Build 126.12] [#436772]

- The rbaOnResponse system parameter fails to work after you upgrade NetScaler ADC nCore or nCore VPX from version 9.3 to 10.x.

  [From Build 129.22] [#480639]

- NetScaler ADC fails to run the commands that have arguments accepting string values and starting with a hyphen (-).

  For example, NetScaler ADC fails to run the following command because the expected value is a string for uat argument that begins with a hyphen.

  bind policy patset ps_adi_any_robots_deny -uat -index 1

  [From Build 131.11] [#508618, 508815]

- The output of the "show cmp parameter" command incorrectly displays the label as "Disable External Cache" instead of "Enable External Cache".

  [From Build 126.12] [#456734]

- If you use the configuration utility to view a Responder action, the Responder Actions page is reloaded.

  [From Build 118.7] [#369583]

- You cannot configure a GSLB service for which a server is not configured on the NetScaler appliance. The configuration utility displays the message "Server must be specified".

  [From Build 118.7] [#360163]

- When search results do not fit onto one page, duplicate records might appear on the second and subsequent pages.

  [From Build 118.7] [#369900, 252063]

- On NetScaler appliances that run the cluster OS, user-defined control policies are not listed in the control flow and therefore do not appear in the Policy Manager. After these policies are bound to Global or an appropriate bind point, they are listed in the data flow.

  [From Build 118.7] [#387554]

- The pagination count on the page listing SSL policies that can be bound does not display the correct values.

  [From Build 119.7] [#372535]

- When a NetScaler session expires, a session expiry message appears in the graphical user interface, and the user has to manually enter the IP address or the domain name of the NetScaler appliance in the address bar to log back on.

  [From Build 120.13] [#361970, 387024, 397473, 400307]

- The Traffic Management > Load Balancing > Set up NetScaler for XenApp/XenDesktop wizard, displays an error if more than one service group is bound to the virtual server that is used for load balancing the XenApp/XenDesktop servers, or if more than one service is bound to the service group.

  [From Build 121.10] [#414807]

- When editing the "Xen Farm" settings in the "Traffic Management > Load balancing > Set Up NetScaler for XenApp/XenDesktop" wizard, load balancing configuration is lost if you switch from XenApp or XenDesktop to Both or from Both to XenApp or XenDesktop. This issue is observed only when Web Interface on NetScaler is the integration point.

  [From Build 121.10] [#414760]

- Unable to access ICA connections through the graphical user interface

  [From Build 121.10] [#420349, 414333, 430665]

- When using the "Traffic Management > Load balancing > Set Up NetScaler for XenApp/XenDesktop" wizard, if you configure XenDesktop and later edit the "Xen Farm" settings to have only XenApp, the XenDesktop bound to the Web Interface site of type Xenappservices in not modified. Therefore, published resources of both, XenApp and XenDesktop, are displayed when accessing the Web Interface site through Receivers.

  [From Build 121.10] [#413087]

- When using the "Traffic Management > Load Balancing > Set up NetScaler for XenApp/XenDesktop" wizard, the compression feature is not enabled on the appliance and for the service groups.

  Workaround: Enable compression on the appliance by using the "enable ns feature CMP" command. Also, enable compression for the service groups by using the "set servicegroup <name> -CMP on" command.

  [From Build 121.10] [#409605]

- When you click the "Edit" link to update the configurations specified in the "Traffic Management > Load balancing > Set Up NetScaler for XenApp/XenDesktop" wizard, an error is displayed when you try to apply the optimization settings.

  Workaround: Edit the XenFarm section (no actual changes required), click Continue and then apply the optimization settings.

[From Build 121.10] [#414361]

- If a SureConnect policy is bound to a virtual server and you upgrade the NetScaler appliance to version 10.1, build 120.13, the policy is not displayed when you navigate to "Traffic Management > Virtual Servers > <virtual server name>.

  [From Build 122.17] [#429652]

- When you use the configuration utility to add a new NetScaler IP address or subnet mask, the qwerty keyboard does not allow you to enter a value greater than 249 for the last octet.

  [From Build 122.17] [#431045]

- When you navigate to "System > Diagnostics" and, under "Utilities", click "TraceRoute" and "Run", the utility uses the default value for Packet Length(44) and displays the error message: Packet length must be greater than 47.

  [From Build 122.17] [#430094]

- The NetScaler configuration utility is not compatible with JRE version 7.45.

  [From Build 122.17] [#426594, 426069, 426185, 453470]

- When using the "Traffic Management > Load Balancing > Set Up NetScaler for XenApp/XenDesktop" wizard for the first time, if you cancel the operation, the configurations that you created are not cleared and you cannot access the wizard again.

  Workaround: Do not cancel the wizard during the first setup. If you want to change a configuration, go through the entire flow, click "Done", and then return to the wizard and click the "Edit" link to update the configuration that you want to change.

  [From Build 123.11] [#414431]

- In a cluster setup, globally bound DNS policies are listed multiple times in the "Bind/Unbind DNS Policy(s) to Global" dialog box.

  [From Build 123.11] [#323213, 388012]

- In the NetScaler configuration utility, virtual servers whose names begin with "APP_" or "app_" are not displayed.

  Workaround: Search for the virtual server names with the expressions "*" or "app" by using the search utility.

  [From Build 123.11] [#438216]

- When using the Traffic Management > Load Balancing > Set Up NetScaler for XenApp/XenDesktop wizard, Web Interface on NetScaler does not publish XenDesktop applications if the load balancing virtual server is configured to listen on two XenDesktop servers.

  [From Build 123.11] [#414422]

- If you navigate to "Traffic Management >Load Balancing > Virtual Servers" and click "SSL Settings" under the "SSL Parameter" tab on the "Create Virtual Server" dialogue box, the "Enable Cipher Redirect" check box is enabled by default.

  [From Build 123.11] [#419409]

- A large ns.conf file can make the configuration utility slow to respond. The large file also slows processing of the

following commands:

- show ns runningConfig

- save ns config

[From Build 123.11] [#405303]

- The comparison between the source IP address of the incoming packets and the configured NetScaler host-name address is unsuccessful because of an endian mismatch.

  [From Build 123.11] [#382199, 462580, 463712]

- If the Surge Protection feature is not licensed, you cannot use the configuration utility to modify the global system settings (System > Settings).

  [From Build 124.13] [#439603]

- If you create a monitor by using the graphical user interface and choose the default browse option to select the in-built monitor scripts from the /nsconfig/monitors folder, the folder does not display any scripts to choose..

  [From Build 125.9] [#447077, 460857]

- The configuration utility includes an option to enable Net Profile when you create a StoreFront monitor, but that option should not be enabled for a StoreFront monitor.

  [From Build 125.9] [#449229]

- If you use the configuration utility to create a NetScaler-owned IP address, and provide the OSPF LSA Type1 area value, the Type1 area value is not displayed when you click on the created IP address to view or edit the details.

  [From Build 125.9] [#443850]

- After you set the SSO Domain (single sign-on domain) value, the value is not displayed on the configuration utility when you navigate to Security > AAA Application Traffic > Settings > Change Global Settings.

  [From Build 125.9] [#446549]

- On a NetScaler SDX graphical user interface, an nsroot user cannot change the passwords of other configured user accounts.

  [From Build 126.12] [#460413]

- The System > Cluster > Manage Cluster screen allows a user to create a cluster without providing a Cluster IP address.

  [From Build 126.12] [#448851]

- In the Traffic Management > Load Balancing > Set Up NetScaler for XenApp/XenDesktop wizard, applying the application firewall policies through the Security settings creates an error condition.

  [From Build 127.10] [#403766]

- The Traffic Management > Load balancing > Set Up NetScaler for XenApp/XenDesktop wizard, displays a distorted view of the published resources when you apply the application firewall settings in the Security section.

[From Build 127.10] [#409057]

- For MPX and VPX Netscalers, you can edit ifalias from the Graphical User Interface properly. If you are using Cluster VPX, you can only edit ifalias using the command line interface and not the Graphical User Interface.

  [From Build 127.10] [#446373]

- The configuration utility might display the following error message when you create a monitor by navigating to Traffic Management > Load balancing > Monitors and click Add:

  Error creating view. Model must not be null

  [From Build 127.10] [#473832, 474471, 490291]

- The NetScaler Application Delivery Controller (ADC) and NetScaler Gateway are vulnerable to the arbitrary code execution in a SOAP interface, as described at http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-7140.

  With this fix, the ADC and NetScaler Gateway do not allow a remote attacker to execute arbitrary code.

  [From Build 129.22] [#483340]

- If you bind a content switching policy to a content switching virtual server, an incorrect value appears in the Configure Virtual Server (Content Switching) dialog box. The error is on the CSW tab, in the Hits column under Policies.

  [From Build 129.22] [#475653]

- A NetScaler ADC displays a Java error if you access it by using an sshd connection.

  [From Build 129.22] [#451546]

- If you access the configuration utility through Internet Explorer 9 or 10 and rename a virtual server, a "No such resource" error message appears, even if the rename operation is successful.

  Workaround: Use the mouse to click the OK button instead of pressing the ENTER key on the keyboard.

  [From Build 129.22] [#374304, 377460]

- The configuration utility displays the "Resource already exists" error if you configure a content switching virtual server with the IP address 10.69.129.128 .

  [From Build 129.22] [#490142]

- If you bind a load balancing monitor to a load balancing service, the Configure Service dialog box displays an incorrect value for response time on the Monitor tab.

  [From Build 129.22] [#488748]

- If a connection from a client to a NetScaler ADC is closed without the client logging out, the session created for that connection remains active until the configured timeout period lapses. If this occurs frequently, after about the 20th occurrence the user might get a "Connection limit to CFE exceeded" error message.

  [From Build 130.13] [#511565]

- The configuration utility does not display SSL policies if you navigate to Traffic Management > SSL > Policies to create a

policy.

Workaround: Navigate to Traffic Management > SSL and, in the right pane, select SSL Policy Manager. Or click the refresh button on the top right corner to display the SSL policies.

[From Build 130.13] [#489884]

- If you have assigned an SSL chip to a VPX instance provisioned on an SDX appliance, you cannot enable or disable TLS1.1 and TLS1.2 protocol support on a virtual server by using the configuration utility.

  [From Build 130.13] [#496957]

- If a NetScaler connection from a client is closed without the client logging out, the session created for that connection remains active until the configured timeout period elapses. If this happens frequently, after about the 20th occurrence the user might get a "Connection limit to CFE exceeded" error message.

  [From Build 130.13] [#375277, 322602, 334465, 396405, 412455, 419503, 438382, 438534, 438796, 441853, 446387, 448361]

- If the number of interfaces that you created are more than eight, the Reporting tab in the configuration utility displays only eight interfaces to be monitored.

  [From Build 130.13] [#494804]

- If you create a GSLB service by using a server name with alphanumeric characters, the server name does not get converted to a server IP address, and the server IP address value is null. As a result, GSLB synchronization fails.

  [From Build 130.13] [#501644, 505641, 509379]

- The statistics of service group members do not appear correctly in the configuration utility.

  [From Build 131.11] [#521579, 508630, 519918, 521983]

- Load balancing virtual servers that are used by AppExpert applications are displayed in nodes other than the AppExpert node. For example, they are displayed in the Available Virtual Servers list in the "Create Persistency Group" dialog box (Load Balancing > Persistency Groups > Add) and in the Target Load Balancing Virtual Server list in the "Create Content Switching Action" dialog box (Content Switching > Actions > Add).

  [From Build 131.11] [#353015]

- The NetScaler configuration utility displays the following error message if a user with no shell access logs on to the NetScaler appliance: "Not authorized to execute this command".

  [From Build 131.11] [#524143]

- Although the default value of the sslv2redirect parameter is "Disabled," the configuration utility incorrectly shows this value as "Enabled" for a new SSL virtual server.

  [From Build 131.11] [#529177]

- The NetScaler configuration utility displays the following error message if a user with no shell access logs on to the NetScaler appliance: "Not authorized to execute this command".

  [From Build 131.11] [#522511, 517993]

- When you use the configuration utility to create a certificate, an error message appears even if the validity period specified is within the acceptable range.

  [From Build 131.11] [#420736, 536924]

- If you configure a command policy for a system user (System> User Administration > Users > <username> >Edit > Insert) by using the NetScaler configuration utility, the check-boxes do not function as expected on the Command Policies screen.

  [From Build 131.11] [#522654]

- NetScaler authentication fails if you use special characters such as & or ; in the password.

  [From Build 132.8] [#542557, 542644, 544420, 547508]

- If you specify the service type as DNS and select the DNS64 and ByPassAAA check boxes, and later navigate to some other service type (for example HTTP), the checkboxes are grayed out because they do not apply to an HTTP service but are not cleared. That is, DNS64 and ByPassAAA are disabled but not set to the default value.

  [From Build 132.8] [#538163]

- Java Runtime Environment (JRE) does not work on Internet Explorer version 10.

  Workaround: Press F12 and set the Document Mode and Browser mode to Internet Explorer 9.

  [From Build 132.8] [#482135]

- If a user with read-only permissions opens a monitor (Configuration > Traffic Management >Load Balancing> Monitors), the configuration utility displays the 'Not authorized to execute this command' error message.

  [From Build 130.13] [#512427]

- If a content switching virtual server with a large number of existing connections is removed, flushing all the PCBs takes time. If any traffic destined for the virtual server is received during this time, the appliance fails.

  [From Build 122.17] [#394856, 353736]

- In a cluster environment, if you run the bind cs vserver command with the argument type, the NetScaler appliance incorrectly reports a difference between the running configuration and the saved configuration.

  [From Build 123.11] [#411116]

- Rebinding a content switching policy to a content switching virtual server might result in memory corruption, which might cause the NetScaler appliance to fail.

  [From Build 123.11] [#432272, 409948, 467208]

- The NetScaler appliance fails in the following scenario:

  1. Create a content switching virtual server (CS1) and bind a policy (P1) to it.

2. Rename the virtual server (CS1) to CS2.

3. Create another content switching virtual server named CS1 and bind P1 to the new CS1.

4. Send traffic to virtual server CS1.

[From Build 125.9] [#428991]

- You must bind only a load balancing (LB) virtual server as the default or target LB virtual server to a content switching (CS) virtual server. Global server load balancing (GSLB), cache redirection (CR), virtual private network (VPN), and CS virtual servers must not be bound to a CS virtual sever as the default or target virtual server.

  [From Build 125.9] [#449261, 451077]

- If an HTTP content switching virtual server is bound to an SSL virtual server that has a backup SSL virtual server, the following error message appears:

  ERROR: The backup vserver of the target vserver is not compatible with the CS vserver.

  [From Build 125.9] [#445561]

- The output of the "stat cs vserver ?fullValues" command now displays the number of requests per second. In earlier builds, the output displayed the total number of requests.

  [From Build 127.10] [#460259]

- If an invalid HTTP request that spans multiple TCP segments is sent to a content switching virtual server, the NetScaler ADC might skip the load balancing decision and initiate a connection from the SNIP address to the content switching virtual server. This can cause the ADC to fail.

  To prevent this problem, the ADC closes the client connection when this situation arises.

  [From Build 130.13] [#501856]

- If you perform the following sequence of actions, the second command fails when the restart process runs the commands, because that process adds the gotopriorityexpression to the second binding:

  1. Bind a policy to a content switching virtual server and specify a gotopriorityexpression.

  2. Bind a filter or compression policy to another content switching virtual server without specifying a gotopriorityexpression.

  3. Save the configuration and restart the appliance.

  [From Build 131.11] [#523636, 532832, 533690]

- The NetScaler appliance, configured to function as DNS forwarder or DNS resolver, may becomes unresponsive whenever it receives UDP DNS truncated response from a name server.

  [From Build 120.13] [#401451, 406480, 409029]

- In DNSRewrite Policy, CLIENT.IP.SRC.MATCHES_LOCATION is an incorrect expression for a response from the DNS.

NetScaler does not recognize this expression and hence might crash.

[From Build 123.11] [#426093, 452776]

- NetScaler caches partial response in the following two conditions:

  1. When the response contains more number of resource records for same domain than the limit mentioned in documents. In such a condition, NetScaler caches response till the maximum limit.

  2. When the response contains invalid RDATA, for example, 0.0.0.0 in address record (A record). In such a condition, NetScaler caches resource record till the invalid resource record.

  In such conditions, when NetScaler received a query for the same domain, it replied with a partial response. Going forward, NetScaler will not cache partial response and in such conditions the queries are directed to the back end server.

  [From Build 123.11] [#385524]

- The NetScaler appliance might fail in the following set of circumstances:

  * On the appliance, you have configured DNSSEC offload and enabled NSEC record generation for a zone.

  * The appliance receives a DNS NODATA/NXDOMAIN query for that zone, over TCP, and the DNSSEC OK bit in the query is set.

  [From Build 124.13] [#376662]

- Statistics do not appear correctly for a DNS load balancing virtual server.

  [From Build 128.8] [#462862]

- CNAME Record Caching

  NetScaler ADC when deployed in a proxy mode does not always send the query for an address record to the back-end server. This happens when for an answer to a query for an address record, a partial CNAME chain is present in the cache. Under few conditions, ADC caches the partial CNAME record and serves the query from the cache.

  For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-5-map/ns-tmg-dns-caching-cname-record-con.html

  [From Build 128.8] [#422509]

- The DNS cache entries are not flushed if the DNS caching feature has been disabled for approximately 250 days.

  [From Build 129.22] [#471707]

- If a server sends a NODATA response that has CNAME record in the answer section and no records in the authoritative and additional sections, the response is marked for CNAME caching on the NetScaler ADC, because it is incorrectly assumed to be a referral response. As a result, the ADC sends a blank response to subsequent queries, of any query type, for the canonical name.

  [From Build 129.22] [#477552]

- When a NetScaler ADC is deployed as a DNS server with caching enabled, and "flush dns proxyRecords" is used when the ADC is serving a large volume of traffic and has a large number of records in its cache, the ADC might fail.

[From Build 129.22] [#484069]

- If the number of records in a DNS response for a domain exceeds the Netscaler ADC limit, or if one of the records in the response contains invalid data, the NetScaler ADC does not cache the response. As a result, DNS resolution using NetScaler nameserver entities fails.

  [From Build 130.13] [#437529]

- If a MySQL client sends a query larger than 16 MB, the query is split into multiple MySQL packets. Only the first MySQL packet in a query is forwarded to the server, and the remaining packets are accumulated on the appliance. After some time the window size is reduced to zero and the client cannot send any more packets to the appliance.

  [From Build 123.11] [#433383]

- A pluggable authentication request causes the handshake to fail. A NetScaler ADC does not support pluggable authentication requests. The flags that indicate pluggable authentication requests are now ignored and the request is processed.

  [From Build 124.13] [#441162]

- NTLM authentication is now supported on all Windows clients.

  [From Build 126.12] [#451036]

- Support for SQL Server High-Availability (HA) Group Deployment

  The NetScaler ADC now supports AlwaysOn Availability group deployment in database specific load balancing for MSSQL 2012.

  For more information, see http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-5-map/ns-dbproxy-db-specific-lb-for-mssql-2012-tsk.html

  [From Build 127.10] [#415485]

- If a service group is used to load balance MSSQL servers that require Kerberos Constrained Delegation, the NetScaler ADC fails to use the proper service port to fetch tickets.

  [From Build 129.22] [#479472, 501750]

- If you use SQL server driver for SQL Server 2000 SP1, the databases are not enumerated for Kerberos authentication on the NetScaler ADC, because the ADC does not process the SSPI packet correctly.

  [From Build 130.13] [#507709]

- The PDF format of NetScaler product documentation is no longer packaged with the NetScaler MPX, VPX, and SDX software. NetScaler product documentation is available in HTML format on the eDocs product library web site. You can generate a PDF for any topic from eDocs.

  To access NetScaler documentation on eDocs, see http://support.citrix.com/proddocs/topic/netscaler/ns-gen-netscaler-wrapper-con.html.

[From Build 118.7] [#395277]

- The configuration utility procedures in the NetScaler 10.1 documentation have not been updated to reflect the new top-level nodes.

  See http://support.citrix.com/proddocs/topic/ns-rn-main-release-10-1-map/ns-rn-changes-gui-10-1-con.html for information about the new node structure.

  [From Build 122.17] [#370607]

- On a NetScaler appliance that has both a monitor and a GSLB view bound to a GSLB service, occasionally the view binding is not visible from the command line and is not saved in ns.conf, even though the GSLB service is properly configured and UP.

  [From Build 118.7] [#394328, 406300]

- In a GSLB setup, if you perform auto synchronization and the configuration file in your local site contains the "add locationFile" command, the command is not synchronized to the remote location.

  [From Build 119.7] [#385305]

- When GSLB virtual server is configured with RTT or Static Proximity as load balancing method or SOURCEIPHASH as the persistence type, NetScaler may reboot because of invalid memory access.

  This issue is observed on the MPX 7500 appliance.

  [From Build 121.10] [#421837]

- If a configuration has a large number of GSLB services and the add location file command is used to add the location database, some of the services might not be assigned a location from the database.

  [From Build 121.10] [#408374]

- On a NetScaler appliance that has GSLB configured, if you remove custom location entries from the GSLB database, the appliance crashes.

  [From Build 123.11] [#413367]

- You can add a GSLB site IP address with a Traffic Domain setting, but this configuration is not supported, and the NetScaler fails. With this fix, you cannot add a GSLB site IP address with a Traffic Domain setting.

  [From Build 124.13] [#434660]

- GSLB static proximity stops working, if you remove the custom records after the database ideal times out. If you have not removed the custom records, then it starts to work when a new connection request is made.

  [From Build 127.10] [#465500]

- In rare cases, high management-CPU usage occurs and a large number of error messages appear in the log file. As a result, queries to the location database might fail, and the backup load balancing method is used for site load balancing.

  [From Build 129.22] [#453144, 455417]

- If you change the GSLB configuration while the GSLB feature is disabled, the NetScaler ADC might process some stale messages when you enable the feature. As a result, the ADC might dump core and restart.

  [From Build 130.13] [#485811]

- The NetScaler ADC fails if a VPN session action, a WI home page, or DBS services are configured with a domain name that at the same time is managed by a GSLB virtual server configured with static proximity or RTT load balancing methods.

  [From Build 131.11] [#433094, 469937, 517974]

- If you force synchronization of the GSLB configuration, the non-default settings on the RPC node are lost. As a result, the GSLB auto-sync functionality is lost.

  [From Build 131.11] [#497412]

- The show gslb service command now displays the following values related to the GSLB service:

  -Last State Change

  -Time since last state change

  -Client and Server idle timeout

  [From Build 131.11] [#498854]

- If the length of the domain name bound to a GSLB virtual server exceeds 31 characters, the domain name is displayed as HASHED STRING during an SNMP MIB Walk operation.

  [From Build 131.11] [#511878]

- All GSLB features except DNS views, auto sync, and static proximity are supported for IPV6.

  [From Build 131.11] [#519589]

- If the disablePrimaryOnDown parameter is configured on the primary GSLB virtual server, the primary GSLB virtual server remains in the DISABLED state even after its health state is UP. The backup GSLB virtual server continues to serve the traffic until HA failover, or until you manually enable the primary GSLB virtual server.

  [From Build 131.11] [#517961]

- A NetScaler appliance in a GSLB configuration might fail if the public IP address of a GSLB service is different on two GSLB sites and, on one of the sites, the public IP address for that service is the address of a load balancing virtual server.

  [From Build 131.11] [#505932]

- If a spillover policy is bound to a GSLB virtual server of type UDP, the show ns runningConfig command does not display the policy binding. The policy binding functions properly, but the configuration might be lost if a failover occurs or if the appliance is restarted.

  [From Build 132.8] [#528060]

- If you enable NTP synchronization on a NetScaler ADC, the ntpd service binds to port 3010. The binding causes resource conflicts, because the port was reserved for the nsnetsvc service.

  [From Build 130.13] [#502309, 503357]

- A user session is not terminated if the user logs out of NetScaler ADC by using the configuration utility. The session is terminated only after the session timeout is complete.

  [From Build 130.13] [#513132]

- On a NetScaler SDX appliance or NetScaler VPX instance, if you use the graphical user interface (GUI) to modify the high availability (HA) monitoring or any other property, the GUI displays the Operation not Permitted error message.

  [From Build 130.13] [#495067]

- In a high availability configuration, on a connection to an FTP virtual server with the stateful connection failover option enabled, if the FTP control connection is closed before the passive mode FTP data connection is opened, the secondary node might become unresponsive.

  [From Build 121.10] [#357841, 408502]

- The synchronization of files in an HA setup stops working after the nsinternal user is disabled.

  [From Build 122.17] [#420089, 409307, 425486]

- On the secondary node of a high availability (HA) configuration, if the HA propagation and HA synchronization options are disabled and Stay secondary is enabled, you cannot disable the Stay secondary option after upgrading the node.

  [From Build 124.13] [#416573]

- On a HA setup, even though the source IP is not explicitly set to *, the output of the "show ns rpcNode" commands shows the source IP as *. Therefore, when HA failover happens for the second time, the LB persistency session information is not propagated to the secondary node. This means that the information is not available when a forced failover is performed on the new primary node.

  The fix ensures that the NetScaler IP (NSIP) address of the local box is always set as the source IP address in a HA setup.

  [From Build 129.22] [#469857]

- In a high availability configuration, if the diff ns config command includes the -ignoreDeviceSpecific parameter, the command fails and does not display the difference in configurations between the two nodes.

  [From Build 131.11] [#524146, 526699]

- If the link between the primary and secondary appliance is very slow and there are a large number (millions) of sessions to be synchronized (because of, for example, load balancing persistence), the primary appliance quickly consumes all the NetScaler memory available for buffering. The lack of buffer space for other subsystems can result in various disruptions, such as failover.

  [From Build 131.11] [#519085, 525203, 533671, 534616, 537991, 539518, 541525]

- In a high availability configuration, if a NetScaler packet processing engine (NSPPE) fails on the primary node, both the

nodes might go into a warm reboot loop.

[From Build 132.8] [#479666, 507519, 541503]

- After an HA configuration is stabilized from a "spilt brain" condition (both nodes primary), connections are not immediately synchronized between the current primary and the current secondary node. This latency might result in an HA failover.

  Workaround: After the HA pair is stabilized, perform a forced synchronization, on either the primary or the secondary node.

  To perform a forced synchronization use the following command:

  force ha sync

  [From Build 132.8] [#537496]

- In a high availability configuration, with failSafe mode enabled on the secondary node, the node might briefly become primary when restarted.

  [From Build 132.8] [#534795]

- The NetScaler appliance fails while processing ICA traffic if you have disabled AppFlow logging on the VPN virtual server (set vpn vserver -appflowlog disable).

  [From Build 122.17] [#417274]

- When NetScaler Gateway is deployed in a double hop setup, the NetScaler fails while processing the packets.

  [From Build 123.11] [#429280, 449953, 463668]

- During an ICA handshake, the version-length value that Mac receiver sends in UNICODE format is parsed incorrectly.

  [From Build 123.11] [#432039]

- The NetScaler Insight Center dashboard displays incorrect Init Program and Client Version values for MAC or HTML receivers on different platforms.

  [From Build 123.11] [#433180]

- The NetScaler Gateway fails if AppFlow is enabled or disabled during ICA connections. The NetScaler Gateway might also fail if the NetScaler appliance receives an ICA parsing error.

  [From Build 123.11] [#430696]

- The HDX Insight console displays unnecessary ICA user-session information and console

  messages.

  [From Build 123.11] [#433511]

- On the NetScaler Insight Center dashboard, the source IP address displayed in the application launch records is incorrect.

[From Build 123.11] [#397109]

- The NetScaler ADC might fail if the EUEM channel data that is part of the ICA traffic flow is split across multiple frames in such a way that the first frame contains only 1 byte.

  [From Build 125.9] [#445959, 451775]

- With some WYSE clients, NetScaler ADC fails while processing the ICA connections if the ICA frame is fragmented across several CGP frames (more than three 3 frames).

  [From Build 125.9] [#445550]

- When appflow is enabled, Multi-Stream ICA connections do not work if an appflow policy is bound to a VPN virtual server and appflow logging is enabled on the VPN virtual server.

  [From Build 128.8] [#458122]

- Once the memory limit for a content group is reached, the memory of the resulting object flush is not handled properly. As a result, no objects are stored after the content groups memory limit is reached.

  [From Build 123.11] [#434877, 436298, 451148]

- The NetScaler appliance fails to respond when it receives multiple byte-range requests for the same objects at almost the same time and where the starting range of byte-range is greater than 1MB.

  [From Build 125.9] [#427598, 446526, 447867]

- When refreshing a cache object for a conditional GET to an expired object, the memory is deducted two times but is returned only once when the cache cell goes away. This causes the memory that is used for a content group to slowly increase and finally reach the maximum memory that a content group can use. The NetScaler appliance is therefore unable to cache objects for that content group.

  [From Build 125.9] [#436298]

- While revalidating cached objects, the integrated caching feature performs some incorrect accounting of the cache size. This causes the NetScaler appliance to crash.

  [From Build 127.10] [#466452, 469584, 469588, 470925]

- The output of the "stat cache -d" command displays an incorrect value for the utilized memory parameter.

  [From Build 128.8] [#427479, 463589, 482725, 502413]

- With integrated caching enabled, the NetScaler can crash when the evaluation of a callout 'result expression' (configured with the resultExpr parameter) results in a UNDEF condition.

  [From Build 129.22] [#488145]

- When you add a new server to an existing service group, the services in the group might be designated as DOWN even though monitoring probes succeed. To enable the services, unset the virtual server spillover method. They are then

correctly designated as UP.

[From Build 118.7] [#391273, 370416]

- Occasionally, when you create a new load balancing virtual server in the configuration utility, a series of error messages appear. The message indicates that the load balancing feature is not licensed, and you are unable to create the virtual server.

  Workaround: Use the NetScaler command line to create the virtual server.

  [From Build 118.7] [#387253]

- If a virtual server is UP because the service(s) are in Transition Out-Of-Service (TROFS) state, the clients do not respond due to requests being queued at the virtual server rather than at the services. Instead, the client must issue 503 or RST.

  [From Build 119.7] [#383402]

- If you change the load balancing group of a virtual server that has a large number of SSL sessions, the appliance might fail.

  [From Build 119.7] [#351870, 399978]

- If you unbind a load balancing (LB) monitor from its service, all the connections to the configured destination IP address (destip) and port (destport) of the LB monitor are closed. In a typical L3 direct server return (DSR) deployment, the destip address and destport of the LB monitor are actually the IP address and port of the virtual server. Therefore, in a typical L3 DSR deployment, if you unbind an LB monitor from its service, all the existing connections to the virtual server are closed. As a result, performance temporarily decreases. The same behavior occurs if you delete a service.

  [From Build 120.13] [#409028]

- In an interactive voice response (IVR) setup, the option selected by a user is not communicated to the server because the RTSP packet is corrupted. As a result, the user is repeatedly asked to select an option from the same list.

  [From Build 120.13] [#390545]

- Monitoring of StoreFront servers fails if they are part of a cluster and the StoreFront monitor is bound to the entire service group. The StoreFront monitor probe fails because individual members have different host names.

  Workaround: If the StoreFront servers are part of a cluster, Citrix recommends that you add them as individual services instead of as members of a service group.

  [From Build 120.13] [#398327]

- Oracle database monitor fills the console window with DONE and DEEP_FLD_LEN messages.

  This issue is observed on the MPX 9500 appliance.

  [From Build 121.10] [#417101]

- If a diameter packet is received by a diameter load balancing virtual server on which persistency is enabled, and that packet contains multiple full requests and a partial request, the NetScaler fails to recognize the partial request and sends it to the server. The result is an invalid packet being sent to the server, and the NetScaler sends a 5xxx message to the client.

[From Build 121.10] [#410711]

- If you run a custom health monitoring script that does not require an argument, the NetScaler appliance sends an incorrect timeout to the script. As a result, the script continues to run for longer than expected. After some time, the maximum limit for the number of scripts allowed on the appliance is reached and new scripts cannot be run.

  [From Build 121.10] [#409055]

- In a high availability setup, after you upgrade the secondary node and make it the new primary, the process of file synchronization from the new secondary (old primary) node to the new primary node overwrites some of the updated data on the new primary. Specifically, the new monitoring scripts delivered as part of the upgrade on the new primary node are overwritten. As a result, the monitoring scripts might fail.

  [From Build 122.17] [#417630]

- In some cases, if you configure a domain-based IPv6 service on the NetScaler appliance, the appliance might become unresponsive.

  [From Build 122.17] [#399446, 416718]

- The stat servicegroup command incorrectly displays the svrttfb (server-time-to-first-byte) value as zero.

  [From Build 122.17] [#424780]

- If the first octet of the IP address of a service has a value of 6 (6.x.x.x), and the service is bound to a virtual server that is configured for persistence, the NetScaler appliance fails when it tries to direct a request to that service.

  [From Build 122.17] [#393613, 427971, 456281]

- If a NetScaler appliance responds to a DNSSEC-enabled request from its cache, and this response is immediately followed by a response from the server to an earlier query that could not be addressed from the NetScaler cache, the appliance drops the response from the server instead of forwarding it. However, the memory associated with the response packet is not freed. As more such requests are received, the memory on the appliance is gradually exhausted.

  [From Build 122.17] [#412530]

- The NetScaler appliance might fail while processing an NX domain message if you have configured an autoscaling service group on the appliance.

  [From Build 123.11] [#402996, 405475, 407313]

- If you use NITRO to display the details of the load balancing monitors configured on a NetScaler appliance, the output for non-HTTP type monitors incorrectly displays a response code, user name, and password. These attributes are not applicable to non-HTTP type monitors.

  [From Build 123.11] [#410365]

- If you create a service of type SSL_BRIDGE and enable client IP address (CIP) on the service, the NetScaler appliance inserts an HTTP header with the client's IP address as its value. In an SSL_BRIDGE topology, you must not insert a header. With this fix, the appliance throws a warning and removes the CIP option for an SSL_BRIDGE service while saving the configuration.

  [From Build 123.11] [#438169]

- If you have configured an autoscaling service group on the NetScaler appliance, the states of some of these services are not updated, because command numbers are not updated. For example, a service state might appear as UP although the monitor has marked it as DOWN.

  [From Build 123.11] [#422821, 405467]

- If you bind a content switching (CS) policy to a CS virtual server, specify a load balancing (LB) virtual server as the target virtual server, and then view the LB virtual server details in the configuration utility, the CS virtual server bindings incorrectly appear in the cache redirection virtual server section. However, if you use the command line to view the details of the virtual server (show lb vserver), the details appear in the correct section.

  [From Build 123.11] [#406467]

- If you configure persistence on a virtual server that is configured for link load balancing, the NetScaler appliance might fail.

  [From Build 123.11] [#392542, 418698, 431925]

- The NetScaler appliance fails under the following sequence of events:

  1. An IPv6 domain based service and an IPv6 address based service are configured on the appliance.

  2. Both the services are bound to a load balancing virtual server.

  3. The domain based service is UP when the address based service enters the UP state.

  [From Build 123.11] [#429445]

- If you have configured a DNS auto-scaling service group and run the "show server <server name>" command to display the details of the server bound to this service group, the following incorrect entries appear:

  - an extra entity with an IP address 0.0.0.0

  - mode as POLICY

  - state as DOWN.

  [From Build 123.11] [#398274, 397588, 425221, 434329]

- If Edge mode is disabled, the state of the name-based service group member appears as UNKNOWN, even though the server represented by the service group member is reachable.

  [From Build 124.13] [#417872, 438960, 465030]

- In a high availability setup, if an autoscaling service group with more than 4000 members is removed, failover occurs.

  [From Build 124.13] [#407493]

- If you rename an autoscaling service group, the NetScaler appliance might fail.

  [From Build 124.13] [#421411]

- If you configure an HTTP_ECV monitor with a response string, and the response arrives in multiple packets, the NetScaler appliance might not parse the response correctly. As a result, a monitoring probe to the appliance fails and

services are marked DOWN.

[From Build 124.13] [#433324]

- If you add a new service group, the SOAP API query for the "show servicegroup" command might fail.

  [From Build 124.13] [#429538, 441186]

- If you have added a backup virtual server on release 9.x, the configuration is lost after you upgrade to release 10.1.

  [From Build 124.13] [#440406]

- If a NetScaler appliance receives a request for which a session does not already exist, the appliance creates a session and designates one of the packet engines (PEs) as the session owner. Subsequent requests that belong to that session might not always arrive at and be handled by the owner PE (for example, PE1). If such a request arrives at another PE (for example PE2), that PE (PE2) gets the information from the owner PE (PE1). Now, the cached session is present in PE2 and the owned session is present in PE1. Because of a timing issue, the information in PE1 is cleared before the cached entry in PE2. As a result, different session entries are created for the same client on PE1 and PE2 and source IP persistence might not work correctly.

  [From Build 124.13] [#420827, 434537]

- Support for Fallback to NTLM Authentication

  Currently AAA supports Kerberos authentication only with Datastream Windows Authentication. AAA does not support fallback to NTLM if Kerberos authentication fails.

  [From Build 125.9] [#382693]

- Using Canonical FQDN when Constructing Server SPN

  When performing Kerberos authentication or authorization, instead of accepting the hostname that the user provided in the request, AAA-TM now performs a DNS lookup on the hostname IP, and uses the canonical FQDN for that IP when constructing a server SPN.

  [From Build 125.9] [#441290]

- In direct server return mode, the NetScaler ADC does not send a RST flag to the client after the idle timeout has expired.

  [From Build 125.9] [#452648]

- The configuration for the NetScaler Web 2.0 Push feature is not saved in the configuration (ns.conf) file. As a result, if you run the "show running config" command, the push configuration is not shown.

  [From Build 125.9] [#451670]

- If a user tries to use a long URL (more than 1024 bytes) to access a protected resource for the first time (that is, without a valid cookie), the NetScaler ADC returns a 500 error.

  [From Build 126.12] [#456632]

- If you bind policies in one of the following orders of priority, and then run the "show running config" or the "save config" command, the command runs repeatedly:

* Syslog, nslog, syslog

* Nslog, syslog, nslog

[From Build 126.12] [#441973, 442098]

- When the primary virtual IP address is down and no backup is configured, spillover persistence fails to decrement the session allocation counter. This leads the NetScaler appliance to believe that sessions are alive and therefore reject new client requests.

  [From Build 126.12] [#454497]

- If you add a server with a name that contains an IP address and a string, and then use that server to add a service, the error message "service already exists" appears.

  [From Build 126.12] [#434925]

- In NetScaler deployments where a load balancing virtual server is deployed behind another virtual server, the count of the number of request bytes is inadvertently doubled.

  [From Build 126.12] [#369369, 252157, 438593]

- The NetScaler ADC does not set the mandatory flag in a Route-Record AVP. As a result, some diameter implementations might reject the AVP.

  [From Build 127.10] [#475980]

- In a deployment with multiple MAC-mode virtual servers, some changes in the configuration can result in a MAC-mode virtual server failing to serve traffic. Changes that can cause the problem include:

  - Disabling and enabling the interface through which the MAC of a service is learnt.

  - Removing virtual servers or clearing their configurations.

  - Changes caused by high availability failovers.

  [From Build 127.10] [#471938]

- The NetScaler ADC fails if requests requiring IP fragmentation are forwarded to a virtual server that is configured for sessionless load balancing in IP mode.

  [From Build 128.8] [#478949]

- If a client connection is in the CLOSE_WAIT state, the NetScaler ADC does not send PUSH notifications to the client. However, it reports success to the PUSH server.

  [From Build 129.22] [#489197]

- If you have configured the RADIUS PI expression CLIENT.UDP.RADIUS.ATTR_TYPE(<avp code>) for content switching, rule-based persistency, or the token load balancing method, and you typecast the result of this expression to an integer or IP address by using the expression TYPECAST_NUM_AT / TYPECAST_IP_ADDRESS_AT, the typecast operation fails.

  [From Build 129.22] [#482113]

- In a high availability setup, a failover might disconnect active connections even though stateful connection failover is enabled on the virtual servers.

  Workaround:

  Check the output of the "show rpcnode" command. If it shows an asterisk (*) for the SRCIP parameter, run the "set rpcnode <remote NSIP> -scrip <local NSIP>" command.

  [From Build 130.13] [#489400]

- If a load balancing virtual server on which persistence is configured is bound to a load balancing group that has no persistence setting, the NetScaler ADC does not change the virtual server's persistence setting. As a result, when traffic arrives at the virtual server, it tries to create a persistence session, but that session fails and the number of sessions increases.

  Workaround: Run the "set lb group ?persistenceType" command to reset the persistence on the virtual servers that are bound to the group.

  [From Build 130.13] [#497470]

- If a semantically incorrect command is entered while a domain based service is being resolved to a NetScaler-owned IP address, the NetScaler ADC displays the state of the service incorrectly.

  [From Build 130.13] [#502338]

- A very slow memory leak occurs on the secondary node in a high availability pair if all of the following conditions are met:

  a) The configuration is large (approximately 4MB).

  b) The configuration includes a large number of "bind lb group" commands.

  c) Configuration changes very frequently, resulting in frequent synchronization.

  [From Build 130.13] [#457639]

- You can now bind loopback members (for example 127.0.0.1) to service groups. Previously, you could bind loopback members to services only.

  [From Build 130.13] [#504209]

- Load Balancing

  The NetScaler ADC might fail after you rename a server that is bound to a service group. This problem does not occur if you assign a name to a server that was previously identified by its IP address.

  [From Build 131.11] [#443027]

- The NetScaler ADC might fail if a high idle timeout value is set on a TFTP load balancing virtual server and the ADC runs out of memory.

  [From Build 131.11] [#505543]

- Load Balancing

If your spillover policy contains the ACTIVETRANSACTIONS or the SURGECOUNT expression (for example, <expression>. ACTIVETRANSACTIONS.GT(<N>)), traffic might spill over to the virtual server bound to this policy even though the current value of the counter has not reached N. This is because these two expressions use an arbitrary number for comparison.

For example, spillover to a virtual server bound to the following policy might occur before the active transactions counter reaches a value of 10:

SYS.VSERVER("A').ACTIVETRANSACTION.GT(10) -action spillover

[From Build 131.11] [#516615]

- The SIP monitor probe has an invalid character in the VIA header. As a result, the probe fails and an incorrect service state might appear.

  [From Build 131.11] [#519644]

- Unsetting one of the load balancing virtual server parameters, such as redirect URL, backup virtual server, push virtual server, or authentication profile, incorrectly unsets the appflowLog parameter.

  [From Build 132.8] [#523239]

- A NetScaler appliance that has AAA-TM configured for authentication with a RADIUS Server might intermittently generate "HTTP/1.1 Internal Server Error 6" error messages.

  [From Build 118.7] [#391105, 457607]

- If you attempt to create a KCD service account on a NetScaler appliance or virtual appliance that has AAA-TM enabled and integrated caching disabled, a buffer overflow might load the appliance or cause it to fail.

  [From Build 119.7] [#402472, 397716, 403737, 404942, 465004, 474112]

- On a NetScaler appliance with AAA enabled and configured, a user whose account is bound to over 100 groups might be unable to execute NetScaler commands at the command line despite having the appropriate permissions to do so. To work around this issue, do not bind a single user account to more than 99 groups.

  [From Build 122.17] [#431206]

- On a NetScaler SDX with AAA and SAML enabled and configured, occasionally the NetScaler appliance crashes and generates a core dump during SAML authentication.

  [From Build 122.17] [#426421, 431795, 436267]

- On a NetScaler appliance that has the load balancing and AAA-TM features enabled, a request that contains an extraneous space in the URL might cause the appliance to crash. This issue occurs only with unauthenticated connections; the appliance processes the same request correctly over authenticated connections.

  [From Build 123.11] [#437407]

- On a NetScaler appliance that has the load balancing and AAA-TM features enabled, and that protects an application that uses 401 Basic authentication, if a client authenticates with a browser that does not support cookies, the

appliance might experience repeated crashes or (for HA setups) repeated failovers. The cause is that the appliance does not receive the expected traffic management cookie, fails to reconnect to the existing session, and instead creates a new sesson each time the client connects to a protected resource. If a large number of authentication requests is sent within a short period of time, the abandoned sessions do not expire quickly enough and can therefore consume available memory.

Workaround: You can either require clients to connect with browsers that support cookies, or reduce the session timeout to a very short interval so that sessions expire quickly.

[From Build 123.11] [#431917]

- If two NetScaler appliances in a high-availability configuration have TCPB mode enabled globally, and you create a DNS TCP service, the service might be successfully created on the primary NetScaler appliance but fail on the secondary appliance.

  [From Build 118.7] [#376173]

- On a NetScaler appliance or VPX virtual appliance that is configured for load balancing in an environment that includes a Microsoft SQL server database, if a client sends a large number of long queries to the MSSQL database, the appliance might become unresponsive or fail.

  [From Build 119.7] [#401118]

- On a NetScaler MPX15000 appliance that has the load balancing and responder features enabled, and has a load balancing policy that includes both the SYS.CHECK_LIMIT and HTTP.REQ.BODY statements, a complex cascade of events might cause the appliance to restart repeatedly. To work around this issue, you can either rewrite the configuration to separate the SYS.CHECK_LIMIT and HTTP.REQ.BODY statements into two separate policies, or operate the NetScaler appliance on a single core.

  [From Build 124.13] [#432790]

- MPTCP transactions of a TCP profile with Selective ACKnowledgement and window scaling might not respond.

  [From Build 120.13] [#401105]

- The NetScaler appliance might not respond when TCP buffering and MPTCP is enabled.

  [From Build 120.13] [#399938]

- The NetScaler appliance does not respond when using client IP insertion with MPTCP.

  [From Build 120.13] [#400888]

- Syncookie cannot be disabled on a TCP profile that has MPTCP enabled.

  [From Build 120.13] [#399708]

- The NetScaler appliances does not acknowledge the subflow FIN when it comes with the MPTCP DATA_FIN.

  [From Build 121.10] [#409426]

- While using MPTCP, the NetScaler appliance cannot adequately handle overlapping data sequence maps.

  [From Build 121.10] [#412833]

- While using MPTCP, the NetScaler appliance crashes when trying to free an already freed TCP session.

  [From Build 121.10] [#419184]

- The NetScaler appliance must not send MPTCP control signals such as DATA_FIN or FAST_CLOSE when the NetScaler has already sent a subflow FIN.

  [From Build 121.10] [#414182]

- MPTCP does not support IPv6 addresses.

  [From Build 121.10] [#401793]

- Virtual servers to which a listen policy is bound accept connections from the first subflow only.

  [From Build 122.17] [#400861]

- MPTCP does not support FTP data connections.

  [From Build 122.17] [#400819]

- With USIP enabled, MPTCP requests do not go through.

  [From Build 122.17] [#331338]

- Multiple spillover persistence sessions are created for a single MPTCP transaction.

  [From Build 122.17] [#400875]

- If you bind monitors to services, and then bind a DoS or SureConnect policy to one of those services, save the configuration, and restart the appliance, you lose information about monitors bound to any services created after the service to which you bound the policy was created. Also, if you run the "show ns runningConfig" command before restarting the appliance, the monitor binding information does not appear.

  [From Build 120.13] [#406391]

- A monitor of type CiTRIX-wi-EXTENDED fails if the script name and site path arguments are not explicitly set.

  Workaround:

  1. Create a monitor of type CiTRIX-wi-EXTENDED.

  2. Set the script name.

  3. Set the site path.

For example,

> add monitor wi-mon CiTRIX-wi-EXTENDED -userName administrator -password freebsd -domain xendt -sitePath "/Citrix/XenApp"

> set monitor wi-mon CiTRIX-wi-EXTENDED -scriptname "nswi.pl"

> set monitor wi-mon CiTRIX-wi-EXTENDED -sitePath "/Citrix/XenApp"

[From Build 121.10] [#383812]

- If you bind an FTP user monitor to an IPv6 service, the state of the service is shown as DOWN.

  [From Build 123.11] [#369946]

- Transparent monitors are now combined with the functionality of an ARP monitor. This avoids the need to bind a separate monitor to incorporate reachability as part of the health status. Without an ARP monitor, UP services could not transition to DOWN when the next hop failed.

  [From Build 124.13] [#301570]

- For a service that is bound to a service group, NITRO cannot obtain the state of the service monitor.

  [From Build 123.11] [#424553, 302231, 386570]

- When importing an AppExpert template that has back end services configured, the NetScaler ADC reports a protocol mismatch error even if other service parameters (service name, IP address and port) are not the same.

  [From Build 125.9] [#444986]

- When AppFlow is enabled on a NetScaler, the following query, which requests console messages from nsconmsg tool, results in httpd core dump due to large buffer length.

  http://<NSIP>/nitro/v1/config/clioutput?
  args=command:"shell+nsconmsg+%2DK+%2Fvar%2Fnslog%2Fnewnslog+%2Dd+consmsg"

  [From Build 130.13] [#507594]

- On a NetScaler appliance that has AAA configured with SSL certificate set to "optional" and at least one authentication policy, when Android users attempt to authenticate, the Android Receiver client generates the following error: "invalid server certificate". This error is caused by improper cookie handling by the Android Receiver client.

  [From Build 121.10] [#418200]

- Devices running Windows cannot connect if the NetScaler Gateway virtual server is configured with either TLS 1.1 or TLS 1.2 or both.

  [From Build 125.9] [#445485]

- When users log on, preauthenication might not synchronize between processes. When this occurs, NetScaler Gateway

fails.

[From Build 127.10] [#440623]

- If proxy settings are configured on the user device and the NetScaler Gateway URL is in the proxy bypass list, users cannot establish a VPN connection with the NetScaler Gateway Plug-in for Windows

  [From Build 127.10] [#456179, 462881, 466862]

- If you configure the Green Bubble theme and if users do not meet the domain requirements when changing their passwords, users do not receive an error message. Instead, the logon page appears. With this fix, the error message appears to users.

  [From Build 127.10] [#474027]

- If a configuration change occurs while being referred in the processing engine, NetScaler Gateway fails.

  [From Build 127.10] [#460997, 477547]

- If users connect to a domain-based server by using clientless access, NetScaler Gateway fails occasionally.

  [From Build 127.10] [#412237]

- When users log on with clientless access and then open the Access Interface, the order of files that appear in Personal File Shares differs from the order of files on the file share server.

  [From Build 128.8] [#461225]

- If you disable authentication on NetScaler Gateway, endpoint analysis scan can occasionally be bypassed.

  [From Build 128.8] [#470059]

- If the Domain Name Server (DNS) configuration is not available, users receive an "Internal error 500" message after successfully logging on to NetScaler Gateway.

  [From Build 128.8] [#464956, 470873, 471478, 474012]

- If you bind SAML and LDAP authentication polices to the virtual server for two-factor authentication, after authenticating with SAML which is primary authentication type the LDAP user name populates automatically. If the first logon attempt to LDAP fails, user names are case-sensitive and must be entered again exactly as it appears after SAML authentication. For example, if the user name is populated as JohnDoe@xyzz.com and the user types johndoe@xyzz.com during the subsequent attempt, log on fails.

  [From Build 128.8] [#463871]

- On a multi-core appliance, if session propagation to one core fails, NetScaler Gateway fails.

  [From Build 128.8] [#485042]

- Attempts to connect to the NetScaler Gateway from a Windows-based computer fails with the error 1008 when Transport Security Layer (TLS) block ciphers are configured and TLS 1.2 is enabled on NetScaler Gateway.

  [From Build 128.8] [#468145, 473867]

- When users upgrade the NetScaler Gateway Plug-in from Version 10.1.122.17 or later to the latest Version 10.1 Maintenance Release on a computer that includes an installation of Citrix Receiver, the automatic upgrade fails.

  [From Build 128.8] [#461279]

- If you configure load balancing virtual servers and the Secure Ticket Authority (STA) with the same fully qualified domain name (FQDN), attempts to bind the STA to the NetScaler Gateway virtual server fail.

  [From Build 128.8] [#374296]

- If the authentication server is extremely slow to respond, such as 15-30 seconds or more, this can cause delays with users logging on successfully, even if the amount of simultaneous connections is low.

  [From Build 129.22] [#489343]

- If user names contain a period (.) that have a common prefix before the period, NetScaler Gateway creates cache files based on the prefix. When this occurs, tickets for one user are sent to a different user.

  [From Build 129.22] [#494463]

- In a high availability deployment, if the NetScaler Gateway virtual server is missing on the secondary appliance, NetScaler Gateway fails during session propagation.

  [From Build 129.22] [#481889, 486176, 501408]

- The endpoint analysis scan fails when users log on by using Internet Explorer 11.

  [From Build 129.22] [#417481, 423915, 496637]

- Upgrading to Maintenance Build 122.11 changes the rewrite policy for HTTP.REQ.USER.NAME. This change retrieves the single sign-on name attribute instead of the server logon name.

  [From Build 129.22] [#495610]

- When there are a very large number of simultaneous user authentication requests and the authentication server is slow to respond, Netscaler Gateway can fail.

  [From Build 129.22] [#488182, 489345, 493939]

- When there are a very large number of simultaneous user authentication requests and the authentication server is slow to respond, Netscaler Gateway can fail.

  [From Build 129.22] [#484431, 488182]

- If Kerberos uses x.509 certificates (PKINIT) for single sign-on, NetScaler Gateway fails to obtain tickets if the Key Distribution Center (KDC) returns a realm referral. This can cause the NetScaler Gateway appliance to fail.

  [From Build 129.22] [#484245]

- If users connect to a web resource over Secure Browse and a proxy server resides behind NetScaler Gateway, single sign-on fails. Single sign-on is successful to either the web resource or the proxy server, but not both at the same time.

  [From Build 129.22] [#470013, 480556]

- If the maximum number of users is set to a number greater than 5 on a NetScaler Gateway virtual server, if you remove the Universal license, the virtual server configuration is also removed.

  [From Build 129.22] [#447452, 486009]

- If you configure SAML authentication with signed SAML assertions, if the user connection disconnects before the SAML response is normalized, NetScaler Gateway fails.

  [From Build 129.22] [#489609]

- In a high availability deployment, when users log on with SAML authentication, the secondary appliance fails over.

  [From Build 129.22] [#490075, 485042]

- If you configure endpoint analysis policies, if the session times out and users do not close the web browser, they cannot log on again.

  [From Build 129.22] [#459149]

- If users connect with the NetScaler Gateway Plug-in for Windows and then attempt to receive a call through a softphone, the call fails.

  [From Build 130.13] [#498679]

- When user connects to a multi-core NetScaler Gateway running out of memory during inter-core communication, NetScaler Gateway fails.

  [From Build 130.13] [#513385]

- When users connect from a web browser and enter their SAML credentials, NetScaler Gateway fails. This occurs when you configure pre-authentication policies and two-factor authentication policies with SAML and LDAP with SAML as the primary authentication type and having a higher priority.

  [From Build 130.13] [#506689]

- On nCore systems, when pre-authentication policies are configured or when an admin session timeout elapses, a core dump may occur when the NetScaler Gateway cleans up the session.

  [From Build 130.13] [#523321, 534178]

- When the Endpoint Analysis is configured, the users are redirected to index.html. Otherwise, a session is created for any arbitrary URL if the authentication is disabled on the NetScaler Gateway.

  [From Build 130.13] [#516257]

- In a double-hop DMZ deployment, if the Receiver connection closes and the connection to XenApp or XenDesktop is in progress, the appliance might fail.

  [From Build 130.13] [#508831]

- When pre-auth is configured on ncore systems, or when Session timeout kicks in, the NetScaler Gateway may fail while cleaning up the session.

  [From Build 130.13] [#528011, 527990]

- When users log on with the NetScaler Gateway Plug-in for Windows, attempts to access internal network resources fail from Windows Metro applications, such as Internet Explorer Metro Mode. This occurs when you configure address pools (intranet IP addresses).

  [From Build 130.13] [#505029]

- Responder or URL transform policies that are bound to the Content Switching virtual server are not applied to connection requests that come through NetScaler Gateway.

  [From Build 130.13] [#495867]

- When ActiveSync clients connect to NetScaler Gateway with "Basic authorization", Gateway fails if credentials in the basic authorization header are invalid.

  ActiveSync clients are supported only with AAA-TM servers on Netscaler.

  Note: This fix was provided in Build 120.x of NetScaler 10.1. However, it missed being included in the release notes of that build.

  [From Build 130.13] [#405138, 405517, 408689, 424539, 429017, 429477, 431645]

- Netscaler Gateway might fail on nCore systems if End Point Analysis is configured or if the configured Session Timeout kicks in.

  [From Build 130.13] [#527990]

- If users do not have administrative rights, the Endpoint Analysis Plug-in installation fails.

  [From Build 130.13] [#506686]

- When users log on with the NetScaler Gateway Plug-in, if the users TCP connection closes and the connection to the internal network through NetScaler Gateway is in progress, the appliance might fail.

  [From Build 130.13] [#500207, 508831]

- If ICA proxy is set to On and you configure authorization policies, when users attempt to connect, NetScaler Gateway modifies the host header to the FQDN of the Web Interface or StoreFront server. When this occurs, user log on fails with the message "Error: Not a priviledged user."

  [From Build 130.13] [#501369, 500311]

- The NetScaler Gateway appliance fails during the device certificate check if AppController is configured on the virtual server.

  [From Build 131.11] [#511805, 532549]

- The NetScaler GUI blocks the creation of a Session Action with a "forced time out" value greater than 255 (256 - 65535). The acceptable range for the "forced time-out" property was increased to 65535 at the back-end, but the GUI does not reflect the same.

  [From Build 131.11] [#535530]

- When a user logs on with the NetScaler Gateway Plug-in, if a Domain Name System (DNS) suffix is configured on the

user device, resolution fails. This occurs if a DNS server is not configured and all of the following are configured on the NetScaler Gateway appliance:

- Authoritative DNS

- DNS address record configured with the host name only

- DNS Suffix

[From Build 131.11] [#459311]

- If the NetScaler Gateway virtual server is behind a proxy server and its fully qualified domain name (FQDN) is not resolvable by the local DNS server, endpoint analysis fails and a "failed sending epaq" error message appears.

  [From Build 131.11] [#522700, 531535]

- In MPX devices, there can be a delay delivering UDP packets from the server to the client in full tunnel mode.

  [From Build 131.11] [#503811]

- Remote users who use the Windows full client/plugin to access Netscaler Gateway can encounter an issue if the Internet Explorer browser has "Automatic Configuration Script" settings configured for Proxy, and the automatic configuration script file is unreachable from the user device at the time of Gateway session establishment. In this scenario, the Windows plugin incorrectly connects to the Proxy server configured in the Manual Settings and fails to establish the session. The expected correct behavior in this situation would be to bypass the proxy and connect to NetScaler Gateway directly.

  Users are affected only if:

  1. They use Windows full client for establishing the gateway session

  AND

  2. They have both Automatic Configuration script and Manual configuration for Proxy in their Internet Explorer settings

  AND

  3. The configured Automatic Proxy script file happens to be unreachable from the user's device (for example the Automatic Proxy script file address is an internal address and not reachable remotely).

  [From Build 131.11] [#531520]

- In a high-availabiltity (HA) configuration, the secondary appliance may fail occasionally due to a duplicate free-attempt of a AAA context.

  [From Build 131.11] [#531956, 538937]

- Java Runtime Environment (JRE) version 7, update 51 or later, displays a security warning when NetScaler Gateway for Java is launched. In some cases, JRE blocks the launch.

  [From Build 131.11] [#491076, 535339]

- If existing AAA sessions exist on a Secondary Netscaler after failover with no associated vpn vservers, then the secondary Netscaler can fail during session sync from Primary.

[From Build 131.11] [#529205]

- If a user installs Microsoft Security Bulletin MS14-080 (KB3025390) for Internet Explorer 11, then uses the IE 11 browser to log on to a NetScaler Gateway virtual IP with endpoint analysis, either as pre-authentication or post-authentication check, the endpoint analysis fails and a Download or Skip Check button appears in the browser.

  [From Build 131.11] [#527757]

- The NetScaler appliance crashes under the following conditions:

  - An external service is added with the same IP address as wihome

  - There are existing AAA sessions

  - The IP address of this external service is changed and later removed

  The crash happens when a user logs in and launches an app. This is because the http request, which needs to be forwarded to Web Interface/Storefront, accesses the stale server information resulting in the crash.

  [From Build 132.8] [#529296, 540736]

- Memory is increasing gradually every week by some 1-2% . The Customer started observing this issue after upgrade to 10.1 126.12nc. A memory leak of MEM_SSLVPN module is suspected based on the observation.

  [From Build 132.8] [#512356]

- In order to fix this issue, we unbound the cache policy through the XA/XD wizard. The builds that implemented this fix will not bind to the cache policy in the configuration flow.

  But if the box is upgraded from an older build where the cache policy is bounded, it will continue and the removal of that policy is done manually.

  [From Build 132.8] [#545422, 550597]

- If the Authorization Group field (in NetScaler Gateway > Global Settings > Change global settings) is left empty, the NetScaler appliance throws the "String too short" error.

  [From Build 132.8] [#538804]

- The rba module crashes when rba users send incorrect remote addr data. Sanity checks for remote addrlen were added to prevent failure.

  [From Build 132.8] [#539286]

- OS X Yosemite users connecting to VPX NetScaler Gateway will not be able to access internal UDP or ICMP resources. This would not occur with the MPX NetScaler appliance.

  [From Build 132.8] [#538446]

- Applications that use UDP (User Datagram Protocol) or ICMP (Internet Control Message Protocol) on Mac OS X Yosemite (10.10) such as the ones using audio or video streaming, may be unreliable.

  [From Build 132.8] [#515013, 512064, 538446]

- If the HTTP CONNECT request is received on the existing connection to a NetScaler Gateway virtual server for a non-owner core before the session is fully authenticated and established, the NetScaler Gateway may fail.

  [From Build 132.8] [#534326]

- After the VPN tunnel is established, external websites fail to load intermittently under the following conditions:

  - If enable_vpn_dnstruncate_fix nsapimgr flag is set on NetScaler.

  - DNS servers on NetScaler are configured to send negative DNS response for external DNS query.

  - Split DNS is set to both

  [From Build 132.8] [#524028]

- HDX Insight

  The graph is not plotted for user applications. This issue is observed on navigating to "Dashboard > HDX Insight > Users > <username> > <SessionID> > Applications > More" button.

  [From Build 118.7] [#386543]

- HDX Insight

  In the Dashboard > HDX Insight > Applications page, the "Total Session Launch count" displays an incorrect number of sessions launched.

  [From Build 118.7] [#381522]

- Load balancing, content switching or VPN applications that have a space characters in the name cannot be enabled.

  [From Build 118.7] [#392515]

- Web Insight

  In Configuration > Inventory > Application List page, the value of number of applications displayed and total number of applications can be incorrect.

  [From Build 118.7] [#378044]

- NetScaler Insight Center appliance fails to respond.

  [From Build 118.7] [#377737, 369685]

- HDX Insight

  The graph is incorrectly plotted for user applications. This issue is observed on navigating to "Dashboard > HDX Insight > Users > <username> > <sessionID> > Applications > More > <application name>.

  [From Build 118.7] [#385895]

- The HDX Insight node appears even when all NetScaler appliances have only standard licenses. The node must be visible only when a minimum of one appliance has an enterprise or platinum license.

[From Build 118.7] [#391336]

- HDX Insight

  If you uncheck the ICA check-box and save, you will see Appflow enabled but no reports or data will show.

  [From Build 118.7] [#388453]

- HDX Insight

  The introduction displayed when you log in to a new NetScaler Insight Center appliance provides only Web Insight information. It does not provide information related to HDX Insight.

  [From Build 118.7] [#387257]

- HDX Insight

  The text that is displayed on clicking the orange icon besides a metric does not accurately describe the licensing issue.

  Workaround: Click the "Learn More" button to get more information on the issue.

  [From Build 118.7] [#388093]

- Web Insight

  The "Page analysis" button is misplaced and not functional on the Dashboard > Web Insight > URL page.

  [From Build 118.7] [#378652]

- HDX Insight

  You cannot enable Appflow on a VPN application for which you have specified an expression from the drop-down list.

  Workaround: The expression for the VPN application must be manually specified as TRUE.

  [From Build 118.7] [#391477]

- Web Insight

  In the Dashboard tab, in some instances, the breadcrumb navigation does not display any text for labels.

  [From Build 118.7] [#390581]

- The HDX Insight node is not displayed for Enterprise licenses of NetScaler appliances.

  [From Build 119.7] [#400665, 405611]

- During an ICA session, the NetScaler appliance fails to respond when you access it's invalid memory space.

  [From Build 119.7] [#405177]

- During an ICA session, the NetScaler appliance fails to respond due to a NULL pointer access.

  [From Build 119.7] [#403134, 403195, 404097, 405013, 408650]

- The load time and render time metrics are not displayed for standard or enterprise licenses of NetScaler appliances.

[From Build 119.7] [#400900]

- The help page on the Graphical User Interface (GUI) displays incorrect information for enabling data collection.

  Workaround: To view the details, click the help icon on the GUI and when the help page opens, click on the TOC tab and navigate to NetScaler Insight Center 10.1 > Enabling Data Collection.

  [From Build 119.7] [#400545]

- Unable to add the IP address in the inventory which contains the number 255 in any quadrant.

  [From Build 119.7] [#332854]

- During installation of a virtual NetScaler Insight Center on VMware ESX, NetScaler Insight allocates only 14 GB of space in the var directory, even though the OVF file specifies 120 GB.

  [From Build 120.13] [#408495, 410708]

- HDX Insight does not support XenApp versions earlier than 6.5.

  [From Build 120.13] [#414844]

- If the memory usage on the NetScaler Insight Center appliance reaches the maximum limit, the appliance fails to respond to further memory-allocation requests by other modules and becomes unresponsive.

  [From Build 120.13] [#402458]

- On the "Dashboard > Users" page, ICA RTT values displayed on the graph in the left panel do not match the values displayed below the graph, or there is a delay in the updating the values.

  [From Build 120.13] [#405818]

- In certain situations, the NetScaler appliance incorrectly interprets the compression buffer size negotiation between the client and the server, and enabling Appflow on the ICA connection causes the appliance to fail when the connection is used to launch an application or desktop. This problem most commonly occurs when a CloudBridge appliance or any WAN optimization device is placed between the client and the NetScaler appliance.

  [From Build 120.13] [#402959, 413016, 413657, 414382, 419571]

- On the Dashboard > HDX Insight > Users page, the report for user sessions displays incorrect values. The left pane displays the average values for the entire session, but, the right pane displays the values for the period selected from the drop-down list.

  [From Build 120.13] [#397236]

- If a CloudBridge appliance is placed between the client and a NetScaler appliance, and AppFlow is enabled for ICA traffic, the XenApp/XenDesktop applications fail to launch and the NetScaler appliance fails.

  [From Build 120.13] [#415812]

- The Active App count in the left pane of the User Details page is not updated instantly.

  Workaround: The correct value is displayed on the Application Details page.

[From Build 120.13] [#395022]

- In a mixed XenApp/XenDesktop server farm, if the XenApp and XenDesktop versions are older than 6.5 and 5.0 respectively, the applications fail to launch because the NetScaler appliance incorrectly parses the ICA packets.

  [From Build 120.13] [#411107]

- For an Active session, data is sent to the AppFlow collector even if the policy rule is changed to FALSE when the session is active.

  Workaround: Start the session again.

  [From Build 120.13] [#369664]

- NetScaler appliance may fail to respond when AppFlow is enabled on the NetScaler Insight Center and the user tries to access the XenApp/XenDesktop farm.

  [From Build 120.13] [#413016]

- In some situations, the NetScaler appliance fails after parsing ICA traffic incorrectly.

  [From Build 120.13] [#413657]

- If you have installed NetScaler Insight Center virtual appliance on ESX, then the console may display watchdog timeout errors or the Graphical User Interface (GUI) may freeze sometimes.

  [From Build 120.13] [#402727, 406388]

- When you enable HDX Insight on a VPN server and try to launch an application from the XenApp server, the NetScaler appliance might fail as it copies the data to an invalid memory location.

  [From Build 121.10] [#423840, 426203]

- NetScaler appliance might fail if AppFlow is enabled and the user tries to access a XenApp/Xendesktop farm under certain network conditions that result in split data packets.

  [From Build 121.10] [#414137, 410495]

- In some cases, NetScaler Insight Center reports incorrect values for XenApp launch count.

  [From Build 121.10] [#416889]

- On the "Dashboard > HDX Insight > Users" page, the report for a specific user does not display data for the Total Application Launch count.

  [From Build 122.17] [#398844]

- Even when Appflow is disabled for a virtual server, you can still clear the configurations on the NetScaler Insight Center by selecting the "Clear AppFlow Configurations" from the "Action" list.

  [From Build 122.17] [#399329]

- If the values for certain metrics are zero, the graphs display these values incorrectly.

  [From Build 122.17] [#403665]

- The "Total App Launch Count" is not displayed when you navigate to "Dashboard > HDX Insight > Gateways" and display the summary for a particular user.

  [From Build 122.17] [#394613]

- The WAN jitter and DC jitter values are not displayed in the NetScaler Insight Center reports.

  [From Build 123.11] [#412129]

- On the dashboard, the table that appears when you navigate to "HDX Insight > Gateways" might display a blank desktop name.

  [From Build 123.11] [#424610]

- NetScaler entity names are case insensitive, but NetScaler Insight Center expects the virtual server names or policy names to be case sensitive.

  [From Build 123.11] [#405849]

- In certain scenarios, if data sent from the XenApp server to the client receiver is delayed because of network congestion or increased network latency, the client re-transmits the ICA magic string, which causes the NetScaler Gateway to fail. This failure happens because the NetScaler Gateway was not expecting two packets containing the magic string.

  [From Build 123.11] [#437475, 441040, 454436, 456445, 459454, 459455, 465311]

- After the NetScaler upgrade or downgrade operation, NetScaler Insight Center does not report any data on the dashboard.

  [From Build 123.11] [#405936]

- On the "Dashboard > HDX Insight > Users" page, the line-graph plots might not add up to the summary shown to the left of the line graph for average bandwidth.

  [From Build 123.11] [#397258]

- In certain scenarios, if data sent from the XenApp server to the client receiver is delayed because of network congestion or increased network latency, the client re-transmits the ICA magic string, which causes the NetScaler Gateway to fail. This failure happens because the NetScaler Gateway was not expecting two packets containing the magic string.

  [From Build 123.11] [#439088]

- If you add a NetScaler appliance to a NetScaler Insight Center setup while ICA sessions are enabled, NetScaler Insight Center does not report the existing ICA sessions. It reports only the ICA sessions initiated after the appliance is added.

  [From Build 123.11] [#417415, 421148]

- The HDX Insight dashboard displays the host delay as server side server-side NetScaler delay.

  [From Build 123.11] [#439992]

- If you move columns and refresh the page, the column ordering is sometimes reset to default.

  [From Build 124.13] [#414155]

- After you enable appflow on some virtual servers, even though no error message appears, the Insight column does not display a check box indicating that the feature is enabled.

  Workaround: Refresh the screen. If appflow is enabled, the check box in the Insight column is selected.

  [From Build 124.13] [#346171, 333555]

- If a NetScaler ADC is deployed in transparent mode for HDX Insight, Citrix Receiver fails to launch the applications or desktops if the appflow policy is not bound to a global bind point.

  [From Build 126.12] [#452989]

- If a NetScaler ADC is deployed in transparent mode for HDX Insight, Citrix Receiver fails to launch the applications or desktops if use source IP (USIP) is enabled and use subnet IP (USNIP) is disabled.

  [From Build 126.12] [#451609]

- The report for desktop session count also includes the count of XenApp sessions, which are launched by the user.

  [From Build 126.12] [#409885]

- On the Dashboard > Web Insight > Applications page, the report for a specific application does not display the client type and client version details.

  [From Build 126.12] [#456449]

- On an HTTP virtual server, after you enable AppFlow by selecting the expression TRUE and the "HTML Injection" box, if you change the policy expression and disable HTML injection, the rewrite and responder policies are still bound to the load balancing virtual server.

  [From Build 126.12] [#401514]

- On the dashboard, HDX Insight reports do not display the active sessions and also displays an incorrect value for session launch count.

  [From Build 126.12] [#453764]

- On the dashboard, when you navigate to Web Insight > Devices > (device record) and click on HTTP Request Methods, HTTP Response Status, Operating Systems, or User Agents, and then from the bread crumb navigation click Application from the respective drop down list, the graph does not display any details.

  [From Build 127.10] [#450474]

- A memory corruption issue causes a NetScaler ADC with AppFlow for ICA enabled to fail.

  [From Build 128.8] [#459668]

- If you enable AppFlow for ICA traffic on a NetScaler ADC, the NetScaler ADC might fail because of an internal memory re-use and dependency issue.

  [From Build 128.8] [#482748]

- If you enable and then disable AppFlow on a NetScaler ADC, the ADC fails while sending the ICA AppFlow records.

[From Build 128.8] [#474159, 475853]

- A NetScaler ADC fails when it receives ICA traffic from metro receiver client.

  [From Build 129.22] [#475981, 477602, 482413, 485138]

- If you enable AppFlow for ICA traffic on a NetScaler ADC, and if there is a large number of sessions, the ADC might fail because of an internal memory re-use and dependency issue.

  [From Build 129.22] [#486792]

- A NetScaler ADC fails when it receives ICA traffic from metro receiver client.

  [From Build 129.22] [#482413, 492160]

- The NetScaler ADCs being monitored by NetScaler Insight Center might fail if, while ICA sessions are active, you enable AppFlow for ICA and then either clear the configuration or disable and re-enable AppFlow on NetScaler Insight Center.

  [From Build 130.13] [#505985, 507879, 507882]

- The NetScaler ADC might fail if you enable AppFlow for ICA and access XenApp or XenDesktop through the Windows Receiver client.

  [From Build 130.13] [#490680]

- You cannot install an SSL certificate on a NetScaler Insight Center virtual appliance.

  [From Build 131.11] [#541712]

- /var/mps/system_health directory is not created for Insight Center. Because of this the techsupport files are not created for Insight Center.

  Workaround: You can manually create the system_health directory after which the techsupport would work as intended.

  [From Build 132.8] [#494666]

- If the /var/mps/policy/mps_policy_backup.xml file is empty or corrupted, the appliance performs a core dump and the Management Service user interface is blank.

  [From Build 118.7] [#385037]

- If you create a static channel, you cannot use the Management Service to remove more than one member interface at a time from the channel.

  [From Build 119.7] [#400607]

- If you modify a NetScaler instance from the Management Service, binding 1/x and 10/x interfaces to an L2 VLAN fails.

  Workaround: Provision the NetScaler instance again.

  [From Build 119.7] [#400409]

- The SVM restore operation of NetScaler instances fail as the SVM shuts down the NetScaler instances that are still being provisioned.

  [From Build 120.13] [#405921]

- If you use the Management Service to delete a channel on which an L2 VLAN was created, the L2 VLAN setting on the NetScaler instance is not cleared. Therefore, the channel continues to be listed on the "VLAN Settings" page of NetScaler instance "Modify NetScaler Wizard".

  [From Build 120.13] [#399972]

- If, when provisioning or modifying a NetScaler instance, you configure an L2 VLAN on a channel that was created by using the Management Service, the configuration fails.

  [From Build 120.13] [#400502]

- After the SDX appliance restarts, NetScaler VPX instances on the appliance cannot send packets tagged with VLAN IDs through an LACP channel.

  [From Build 120.13] [#410416, 444395]

- SSL certificate installation on a NetScaler instance from the SDX Management Service fails during validation if the SSL certificate does not have an associated key file.

  [From Build 120.13] [#405115]

- When you display the running configuration of a NetScaler instance in the Service Management interface, the double quotation marks (") are replaced with HTML code (;quot&).

  [From Build 121.10] [#413123]

- The format of the APPFW CSRF TAG syslog message is not in the expected format. As a result, Command Center displays incorrect values, under AppFirewall Recent Logs, in some fields for this type of AppFirewall syslog message.

  [From Build 122.17] [#414851]

- The SNMP responses are not as specified by the RFC 4001.

  [From Build 122.17] [#420630]

- If a NetScaler instance is created with a Management VLAN using the 0/1 or 0/2 interface, the guest VMs fail to start after provisioning, because the guest VMs use the VLAN networks instead of physical network while assigning the interface.

  Workaround:

  1.Remove the NetScaler instances whose management ports are in tagged VLAN.

  2. Logon to the XenServer shell prompt and remove all the VLAN networks.

  [root@netscaler-sdx ~]# xe vlan-list

  uuid (RO) : bd0dc3b4-2f9f-4db4-0b2d-c1b891a8caf9

tagged-PIF (RO): 2b62ede6-de58-92b1-a2fa-46cd93dd8268

untagged-PIF (RO): 6ff1bd03-2acc-c035-1f7e-ad57952558d3tag (RO): 100

[root@netscaler-sdx ~]# xe vlan-destroy uuid=bd0dc3b4-2f9f-4db4-0b2d-c1b891a8caf9

3. Create the guest VM instances first, and then create the NetScaler instances.

[From Build 122.17] [#424588]

- If you create a channel on interfaces 0/1 and 0/2 by using the Management Service, and then provision a third-party instance and configure the management network for that instance on the newly created channel, the third-party instance is not reachable on the network.

  [From Build 122.17] [#400651]

- When viewing the built-in or custom reports on the Reporting tab on a NetScaler VPX instance running on the NetScaler SDX 17550/19550/20550/21550 platform, the following message appears: NO DATA TO CHART.

  [From Build 123.11] [#262505, 408110]

- A NetScaler SDX appliance intermittently stops processing traffic on interfaces that are part of an LACP link aggregation interface that is transmitting a small amount of traffic.

  [From Build 123.11] [#434738, 446641]

- Even after you configure a short message service (SMS) server, you do not receive an SMS message when an alert is generated.

  [From Build 123.11] [#430449]

- Deletion of a management channel from the Management Service might not always succeed.

  Workaround: Try deleting the management channel again from Management Service.

  [From Build 123.11] [#433054]

- If a management channel exists on a NetScaler instance, you cannot trace the route of a packet from the Management Service to a NetScaler instance.

  [From Build 123.11] [#431243]

- If you create an LACP channel with more than 8 member interfaces, or a static channel with more that 16 member interfaces, the following error message appears: "Channel Interface String Length: 185 is greater than maximum allowed length:128".

  [From Build 123.11] [#424630]

- Descriptors in the NetScaler SDX SNMP MIB file include underscore characters, which are invalid. Only alphanumeric characters are supported.

  [From Build 123.11] [#430097]

- If you apply a license after modifying the SVM host name, the license application might fail.

Workaround: Reboot the Management Service after changing the host name, and then try applying the license again.

[From Build 123.11] [#431463]

- If you specify secure-only access on a NetScaler instance, single sign-on to that instance from the Management Service user interface is not successful.

  [From Build 124.13] [#396252]

- After you create, modify or delete an LACP channel, one of the member interfaces might temporarily stop transmitting. The NetScaler instance might intermittently show the status of the member interfaces as Error-Disabled (in the command line) or DOWN (in the configuration utility).

  Workaround: Log on to the NetScaler instance and execute the following command on the interface that is shown as Error-disabled:

  "enable interface <interface_id> (eg. enable interface 1/1)"

  [From Build 124.13] [#370574, 431840, 442436]

- If you use the Management Service to bind a new interface to an LACP channel, the member interfaces of the channel are reset. As a result, the traffic is not evenly distributed among the interfaces in the channel.

  [From Build 124.13] [#399630]

- On NetScaler SDX appliance, the NetScaler instances do not start when the total number of interfaces and SSL cores is more than 26.

  [From Build 125.9] [#446985]

- If the administrative password for the Management Service contains an ampersand character (&), communication between Management Service and XenServer is affected, and errors occur during provisioning or modification of the instances.

  [From Build 125.9] [#447773]

- When you click on a NetScaler IP address in the SVM GUI, the NetScaler configuration utility opens without prompting for logon credentials. Log on is done through single sign on (SSO).

  [From Build 125.9] [#456884]

- On a NetScaler SDX appliance running Management Service version 10.1, build 119.7, manually initiated backup operations fail, and a User name missing error message appears.

  [From Build 125.9] [#445598]

- If appliance inventory is going on at the same time when channel is being created, then it may happen that channel is created on the VPX but it is not visible from the SVM.

  [From Build 126.12] [#449247]

- The local storage partition was configured as sda3 instead of sda4 in the disk configuration file for NSSDX-22000 and NSSDX-22000T systems. Installing the supplemental pack 100014 along with the latest release resolves the error in disk

configuration file.

[From Build 126.12] [#455601]

- Management service was showing wrong alert for power supply status with the message that "One of the two power supplies is not working."

  [From Build 126.12] [#460376, 457317]

- If you are using the NetScaler SDX 8015/ 8400/8600 10G platform, no interfaces are shown in the interface list when an LACP channel is being created.

  [From Build 126.12] [#460329]

- When an interface other than 0/1 and 0/2 is being used for management on a VPX and later if that interfaces is made part of a channel creation from SVM, then that channel will not be pushed to this VPX and manual steps will be required to achieve the same.

  A user can delete such channels (made out of data interfaces and used for VPX management) from SVM which will leave the VPX in unmanageable state.

  [From Build 127.10] [#456703]

- Configuring a wrong DNS IP address was slowing internal communication between Management Service and XenServer. With the current release, the DNS look up will be ignored for internal communication.

  [From Build 127.10] [#475099]

- The NSIP modify action from the Management Service results in inconsistent state if the "Save Config" command from the Management Service to VPX takes a long time to respond. This happens because the connection might time-out. The issue has been fixed by increasing the time-out values.

  [From Build 128.8] [#480581]

- Set operation on a channel may lead to channel MAC address becoming zero on a VPX running on an SDX appliance.

  Workaround: Use the set channel <Channel_id> -lamac <mac_address> command.

  [From Build 128.8] [#483430]

- If a management channel modify request is sent through Nitro and a data interface is added in the member interface list, then the request succeeds and makes management channel inconsistent.

  [From Build 128.8] [#481835]

- If a VPX is using an interface A and a channel is created on Management Service using interface A and interface B then this channel should also get added to the VPX. But if the Interface B is already shared to its maximum limit, that is no free VFs are left on interface B then that channel will not be added to the VPX.

  [From Build 128.8] [#436286]

- For a case under the following conditions, when:

  1. A VLAN is present on XenServer on management interfaces (normally ETH0 and ETH1 on most platforms)

2. A management channel created from Management Service is present on SDX, and

3. A VPX is using this management channel.

Then, If the management channel is deleted from Management Service, then post deletion the VPX may be seen with the VLAN present on its management interfaces.

[From Build 128.8] [#482603]

- Management Service gives an error when an SDX administrator tries to bind a management channel while provisioning or modifying a NetScaler instance.

  [From Build 128.8] [#463820, 480347]

- The backup of an SDX appliance was failing with an error "username missing". The root cause for this was that the migration from 9.3.x was failing because of duplicate database entries. Going forward, the Management Service will remove the duplicate database entries resulting in a successful migration.

  [From Build 128.8] [#480054]

- On creating a LACP channel, interface MAC address is altered and the new MAC address will be persistent even after the unbind operation.

  Workaround: On unbind, the interfaces might have same MAC addresses or MAC addresses will all be 0. In such a situation, it is recommended to reboot the VPX.

  [From Build 128.8] [#482122]

- After you unbind the interface from a channel, interface drops the packets sent to the individual interfaces.

  Workaround: In such a situation, it is recommended to reboot the VPX.

  [From Build 129.22] [#484194]

- If you create channels on SDX and use these channels in VPXs and then take a backup of the appliance to restore either the complete appliance or selected instances, then channels are not restored and instances may fail.

  [From Build 129.22] [#432899, 435206]

- In case of shared management of CPU in SDX, licenses fail to load on start-up sometimes if the management CPU is overloaded.

  [From Build 129.22] [#473681]

- The backup file contain more NetScaler instance than allowed instance in the license applied. Now instance restore for a single NetScaler fails with error message "License does not allow more than x NetScaler instance".

  Workaround: SDX appliance should have same license as in the backed up SDX appliance.

  Fix: For instance restore operation, licence validation is done against no of NetScaler selected for restore instead of validating against all NetScaler instance in the backup files

  [From Build 129.22] [#498440]

- On NetScaler SDX 8000 appliances, the Service Virtual Machine (SVM) might not detect the disk correctly, in which case it marks the status of the disk as down in system health monitoring. However, the provisioning of NetScaler VPX instances works as expected. This issue occurs in the following releases:

  - NetScaler 10.1 Build 129.11 or earlier

  - NetScaler 10.5 Build 52.11 or earlier

  [From Build 130.13] [#488794, 497445, 504308]

- Enhancement to add 'lspci -vvvxxx' logging at boot time information to SDX log. It uses logrotate to keep log data from the last 3 boots.

  [From Build 130.13] [#507009]

- The installation of supplemental pack 100015 fails on NSSDX-8200 10G platforms. The root cause of failure is that the install script treats a warning as an error and aborts the installation.

  [From Build 130.13] [#495614]

- The installation of supplemental pack 100015 fails on NetScaler SDX 8200 10G appliances.

  [From Build 130.13] [#502975]

- Restore operation fails when the backup file of newer version is restored in older Management Service version.

  [From Build 130.13] [#502428]

- The management interface of a SDX-8000/SDX8200/SDX-8400 appliance might loose connectivity if the interface is connected to a CAT switch.

  Workaround: Set the speed of the interface to 100 Mbps and disable auto-negotiation.

  [From Build 130.13] [#470002, 460650, 484387, 504145, 505053]

- In SDX NetScaler cluster, SDX management VLAN modifications are not allowed through cluster IP.

  [From Build 130.13] [#469680]

- In Management Service, the Tagall setting configured for channels under Management VLAN settings is not available on VPXs.

  [From Build 130.13] [#506128]

- If a new SSL certificate that requires a key is installed without the key, access to management service GUI is lost.

  [From Build 131.11] [#440208]

- The NetScaler SDX appliance fails if it receives SNMP requests before system initialization.

  [From Build 131.11] [#525871]

- Some of the NIC's may become unusable and may not be visible in Management Service on SDX220XX and SDX241XX platforms running with XenServer 6.1 Supplemental Pack 100016A.

[From Build 131.11] [#536844]

- When a NetSclaer VLAN with tagged option for channels is selected, the native VLAN also gets tagged inside the NetScaler VPX for the channel.

  [From Build 131.11] [#512624]

- When you provision the maximum possible number of VPX simultaneously from Management Service, the Xen Server does not provide the details of correct memory space available immediately. There is a lag in recovering the memory space. For this reason although the memory space is available, you may still get "Not enough memory available" error.

  [From Build 132.8] [#525616]

- In sparse traffic conditions on a NetScaler VPX virtual appliance installed on VMware ESX, some latency might occur in releases after 9.3 as compared to release 9.2. If this latency is not acceptable, you can change a setting on the appliance. At the shell prompt, type:

  sysctl netscaler.ns_vpx_halt_method=2

  Perform a warm reboot for the above change to take effect. To have the new setting automatically applied every time the virtual appliance starts, add the following command to the /nsconfig/nsbefore.sh file:

  sysctl netscaler.ns_vpx_halt_method=2

  [From Build 126.12] [#326388]

- For an IPv6 load balancing virtual server that belongs to a traffic domain, and for which the persistence is set as cookieinsert, the NetScaler appliance does not insert the correct cookie in its response.

  [From Build 118.7] [#359348]

- With Random source port selection for Active FTP enabled on the NetScaler appliance, when an FTP server initiates a connection from the standard TCP port number 20, the NetScaler appliance uses a random port instead of port 20 for the client side data connection.

  [From Build 119.7] [#402068]

- The NetScaler appliance might not send the received IPv6 fragments to the appropriate packet engine for processing, which might result in the NetScaler appliance becoming unresponsive.

  [From Build 119.7] [#402123]

- The Network Visualizer does not display the bound IP addresses of a configured VLAN.

  [From Build 119.7] [#366321]

- If the NetScaler appliance has redundant L2 connectivity with a switch, the NetScaler appliance might mark its link-local IPv6 addresses as duplicate during the DAD (Duplicate address detection)?process.

  [From Build 120.13] [#404861]

- When the conditions specified in an ACL rule include the != operator, the NetScaler appliance might not properly filter packets based on the ACL rule.

  [From Build 120.13] [#401303]

- When IP fragments are received on a load balancing virtual server on which the client timeout?parameter set to zero, the NetScaler appliance might dump core and then restart.

  [From Build 120.13] [#405190]

- The NetScaler appliance might restart if it receives a duplicate IPv6 fragment within a very short time after receiving the original fragment.

  [From Build 121.10] [#404849]

- After unbinding a netprofile from a NetScaler Gateway virtual server, the netprofile cannot be removed from the NetScaler appliance.

  [From Build 122.17] [#416941]

- The NetScaler appliance might become unresponsive when traffic from a TFTP server matches a RNAT rule configured on the appliance.

  [From Build 123.11] [#431652, 454475]

- If you have configured an extended ACL without specifying the optional parameter "source IP address", high CPU spikes might occur when you run the "apply ns acls" command either by using the configuration utility or the NetScaler command line.

  [From Build 123.11] [#424243, 430158, 438766]

- When you reset a member interface of a LACP channel, Tx stalls might increment continuously.

  [From Build 123.11] [#435697]

- If you have configured a TFTP load balancing virtual server with persistency option enabled, the NetScaler appliance might become unresponsive when the virtual server receives some traffic.

  [From Build 123.11] [#428819, 436289, 439158]

- If you have configured more than ten ICMP extended ACLs, high CPU spikes might occur when you run the "apply ns acls" command either by using the configuration utility or the NetScaler command line.

  [From Build 123.11] [#408693]

- For a load balancing configuration in which an IPv6 virtual server is used to load balance IPv6 servers, if the NetScaler appliance processes client's final ACK of the TCP handshake and the first data packets in the same IO cycle, the appliance may not forward the data packets to the server causing the connection to fail.

  [From Build 124.13] [#423856]

- On a NetScaler ADC configured for link load balancing with RNAT, access to external sites fails intermittently.

  [From Build 125.9] [#448738, 453558]

- In a High Availability configuration, if you set the maxFlips, maxFlipTime, or syncvlan parameter of the set HA node command, the NetScaler ADC adds a duplicate entry of the add HA node command to the running configuration.

  [From Build 125.9] [#449175]

- In a high availability configuration, you might lose your VLAN configuration if you upgrade the secondary node to build 125.x from builds: 122.17, 123.11,124.13.

  [From Build 126.12] [#469033, 467726]

- In a high availability configuration in INC mode, net profile and IPset commands propagate to the secondary node.

  [From Build 126.12] [#452434]

- If you have configured active FTP with random source port option enabled for an FTP virtual server, the NetScaler ADC might not handle data connections properly for this FTP server and (NetScaler) might become unresponsive.

  [From Build 127.10] [#477507]

- RNAT configuration might be lost in a NetScaler ADC after you restart it.

  [From Build 127.10] [#475466, 475462, 486447]

- The NetScaler appliance might consume excessive CPU cycles when processing ACL rules.

  [From Build 127.10] [#438557]

- In a high availability (HA) configuration, the secondary node might forward BOOTP and DHCP related traffic using a configured VMAC address instead of interface's MAC address.

  [From Build 127.10] [#457119]

- The NetScaler ADC might not remove the session information of an FTP connection from its memory while closing the connection. When the NetScaler ADC allocates the same memory block for a connection related to a UDP DNS service, the NetScaler ADC becomes unresponsive.

  [From Build 127.10] [#448316]

- The default speed for an LACP channel is set to NONE instead of AUTO.

  [From Build 128.8] [#414407, 485512]

- The NetScaler ADC might fail to evaluate listen policies, containing source or destination ipv6 address/subnet, for certain IPv6 addresses.

  [From Build 129.22] [#496564]

- With more than 1000 IP tunnels configured on a NetScaler ADC, the internal data structure for these IP tunnels might not be updated for some events. This changes the status of these IP tunnels to the DOWN state.

  [From Build 129.22] [#491473]

- In a high availability configuration, run-time information such as service states and load balancing persistence sessions might not propagate to the secondary.

[From Build 129.22] [#441062]

- For a link load balancing with RNAT configuration, the NetScaler ADC might use an incorrect subnet IP (SNIP) address to communicate to the external devices.

  [From Build 129.22] [#480621, 478048]

- The LACP channels of a NetScaler ADC might take around 7 minutes to become functional (UP state) after the NetScaler is restarted.

  [From Build 129.22] [#475622]

- In a high availability (HA) configuration, VMAC configuration might be lost when continuous HA failover happens.

  [From Build 129.22] [#477402]

- The NetScaler ADC might use a large amount of CPU cycles when it receives a burst of GRE traffic, which meets the following criteria:

  - The NetScaler ADC is not the GRE end point for this traffic.

  - The NetScaler ADC creates a NAT session information for this traffic.

  [From Build 129.22] [#480573]

- The NetScaler ADC drops IPv4 packets related to the following protocols:

  - IPv6 encapsulation (41)

  - Fragment Header for IPv6 (44)

  - ICMP for IPv6 (58)

  [From Build 129.22] [#490190]

- The CPU usage might be approximately 10% higher in NetScaler 10.5 version as compared to NetScaler 9.3 version.

  [From Build 129.22] [#432192]

- In a CloudBridge connector tunnel, IKED packets might get routed back to the same NetScaler ADC instead of the peer tunnel end point.

  [From Build 129.22] [#494875, 498447]

- In a transparent cache redirection deployment, when a request is destined to a MAC address (say MAC-A) and the response for the request is sent from another MAC address (say MAC-B), the NetScaler ADC sends further requests to MAC-B. If MAC-B stops handling the requests, the session might get hung.

  [From Build 129.22] [#460246]

- For a link load balancing with RNAT configuration in which persistence is enabled for the virtual server, the NetScaler ADC might become unresponsive when the virtual server receives traffic.

  [From Build 129.22] [#471651, 479882, 485831, 493232]

- On a NetScaler ADC, ND6 entries might get in INCOMPLETE state due to synchronization mismatch among different internal modules. As a result NetScaler fails to serve traffic for that IPV6 address.

  [From Build 129.22] [#480100, 483728]

- The NetScaler ADC might not update its bridge and ARP tables with the information received from GARP messages.

  [From Build 130.13] [#497277]

- For a load balancing server configured on a non-default traffic domain, modifying the IP address of the server also changes the name of the server.

  [From Build 130.13] [#496237]

- Old or stale OSPF LSAs might exist after a warm restart, or a restart after a power failure, resulting in a triple flip.

  [From Build 130.13] [#441005]

- With MAC based forwarding (MBF) enabled, the NetScaler ADC does not update Layer 2 information such as MAC address, interface ID, and VLAN ID, for a dynamic service even when the associated router is inactive. As a result, the router drops the packets destined to the IP address specified by the dynamic service.

  [From Build 130.13] [#490341]

- An Access Control List (ACL) rule specifying the TCP protocol and the Established option might not get evaluated if another ACL rule with a higher priority also specifies TCP.

  [From Build 130.13] [#510173]

- If you disable the TCP Proxy parameter while creating a Reverse Network Address Translation (RNAT) rule on a multi-core NetScaler ADC, the NAT operation fails.

  [From Build 130.13] [#508631, 509453]

- The NetScaler ADC might become unresponsive when ICMP error packets match a forwarding session rule.

  [From Build 130.13] [#502213, 512248]

- If you bind an interface with a unit number greater than 31 to a VLAN that is used as a Sync VLAN in an HA configuration, the Sync VLAN becomes unoperational.

  [From Build 131.11] [#507345]

- In an active-active configuration, services bound to the backup VIP addresses do not send monitor probes to the associated servers.

  [From Build 131.11] [#355965, 485260]

- Upon receiving Generic Routing Encapsulation (GRE) packets as IP fragments on a virtual server with protocol ANY, the NetScaler ADC fails and restarts. This occurs only when you do not explicitly configure a GRE tunnel on the NetScaler ADC.

  [From Build 131.11] [#522538]

- In a high availability (HA) configuration, ACL rules that are configured to block SSH related packets also block HA file synchronization that internally uses the SSH protocol.

  [From Build 131.11] [#438901]

- An ACL6 rule might not get evaluated for a series of TCP packets.

  [From Build 131.11] [#528554]

- An active FTP connection might get reset for no apparent reason, regardless of the state of the random source port.

  [From Build 132.8] [#507908]

- In an active-active configuration with the sendToMaster parameter enabled, the backup nodes might not forward packets to the master node.

  [From Build 132.8] [#554336]

- Blocking Traffic on Internal Ports

  The NetScaler appliance does not block traffic that matches an ACL rule if the traffic is destined to the appliance's NSIP address, or one of its SNIP addresses, and a port in the 3008-3011 range.

  This behavior is now specified by the default setting of the new Implicit ACL Allow (implicitACLAllow) parameter (of the L3 param command). You can disable this parameter if you want to block traffic to ports in the 3008-3011 range. An appliance in a high availability configuration makes an exception for its partner (primary or secondary) node. It does not block traffic from that node.

  To disable or enable this parameter by using the command line interface

  At the command prompt, type:

  > set l3param -implicitACLAllow [ENABLED | DISABLED]

  Note: The parameter implicitACLAllow is enabled by default.

  Example

  > set l3param -implicitACLAllow DISABLED

  Done

  [From Build 132.8] [#529317]

- The NetScaler hardware might sometimes report incorrect values for system health counters. The health counters are read over the SMBus, which is prone to reporting wrong or zero values.

  [From Build 118.7] [#373125]

- In certain cases, error messages on the console of an MPX 5550/5650 or MPX 8200/8400/8600 appliance continuously scroll if the physical registers are not correctly read.

  [From Build 118.7] [#360223, 363330, 368513, 374726, 376201, 383863, 385560, 387301, 388487, 392958, 396159,

417578, 426783, 456228]

- The NetScaler license is not processed if the configuration file (ns.conf) contains multiple instances of the host name, or if the host name in the ns.conf file is different from the host name in the rc.conf file. With this fix, if the ns.conf file contains multiple host names, only the name set by the "set ns hostname" command is used. Also, the host name in ns.conf no longer takes precedence over the host name in rc.conf.

  [From Build 120.13] [#409202]

- On the MPX 22040/22060/22080/22100/22120 appliance, if the 10G ports are not populated, the appliance takes about 20 minutes to finish the restart process.

  [From Build 123.11] [#432687]

- NetScaler does not display the correct daylight savings time for Israel.

  [From Build 123.11] [#428562]

- With recent versions of the ixgbe driver, the dmesg.boot file and the show interface command report that the FTLX1471D3BCV-I3 LR SFP+ port is unsupported. This issue occurs with the following releases and builds:

  - Release 10.1 starting build 112.15 or later

  - Release 10 build 74 or later

  - Release 9.3 build 62.4 or later

  - Release 9.3.e build 59.5003.e or later

  [From Build 123.11] [#410251]

- The MPX 11515/11520/11530/11540/11542 platform now supports NetScaler release 9.3 build 65.x.

  [From Build 124.13] [#395280]

- If you try to form a cluster of MPX 22040, MPX 22060, MPX 22080, MPX 22100, and MPX 22120 appliances, the appliance on which you issue the "join cluster" command performs a core dump and restarts. As a result, that appliance is not added to the cluster.

  [From Build 124.13] [#435200]

- The MPX 22040/22060/22080/22100/22120 platform now supports NetScaler release 9.3 build 65.x.

  [From Build 128.8] [#311561]

- For NetScaler platforms that have Small Form-factor Pluggable (SFP) transceivers, with part number FTLF8519P3BNL, the bootup log files show that the SFPs are unsupported, even though they are functioning properly. This issue occurs in the following releases:

  - NetScaler 9.3 Build 67.5 or earlier

  - NetScaler 10.1 Build 129.11 or earlier

  - NetScaler 10.5 Build 52.11 or earlier

[From Build 129.22] [#501834]

- NetScaler VPX instances running on Xen Server might consume a high percentage of CPU cycles while processing 1G traffic.

  [From Build 130.13] [#498929]

- On a NetScaler ADC that has a Small Form-factor Pluggable (SFP) interface with part number FTLF8519P2BNL, disabling this interface might not disable the interface of the peer device.

  [From Build 130.13] [#487169]

- NetScaler supports Multi-PE for Hyper-V.

  [From Build 130.13] [#484123]

- NetScaler VPX instances running on VMware ESXi lose network connectivity when you apply either of the following patches:

  - ESXi550-201410401-BG

  - ESXi510-201410401-BG

  Workaround: For more information, see http://support.citrix.com/article/CTX200278.

  [From Build 131.11] [#510673, 517241, 538267]

- On NetScaler MPX 22040/22060/22080/22100/22120 and ByteMobile T1200 appliances, SNMP based alarms are supported for only first two power supplies.

  [From Build 131.11] [#525360]

- The user interfaces (command line and configuration utility) of a NetScaler instance running on a SDX appliance do not display the actual state of the management ports.

  [From Build 131.11] [#251216, 302381]

- You can now upgrade the Lights Out Management (LOM) firmware directly from the host, without configuring the LOM port on the network.

  [From Build 131.11] [#430733, 542439]

- If an HTTP callout is configured with a virtual server that has a widcard port, the NetScaler appliance fails to respond the first time the callout is triggered.

  [From Build 119.7] [#391238]

- When a filter policy is globally bound to a NetScaler, application firewall or compression or authorization policies that are bound to a content switching virtual server are not saved in the running configuration. However, these bindings are displayed when you run the "show cs vserver" command.

  [From Build 122.17] [#410624]

- After upgrading to NetScaler 10.1, policies that were globally bound to the NetScaler are also being bound at a virtual server level.

  [From Build 122.17] [#429232]

- The NetScaler appliance might fail to respond in the event that a policy of the form HTTP.REQ.BODY(n).AFTER_STR(target-string) has a large value for "n" (for example, 40000) and when the appliance receives large requests in combination with requests with no content length.

  [From Build 123.11] [#417071, 423206]

- A memory leak in the XML_DECRYPT() policy function can consume all NetScaler memory, making it unavailable for other operations.

  [From Build 124.13] [#442807]

- The NetScaler appliance may fail to respond if it does not have sufficient memory during the execution of an XML_DECRYPT function in a policy expression.

  [From Build 124.13] [#414552, 429079]

- Error messages displayed during policy binding are shown as hexadecimal code instead of the corresponding warning message.

  [From Build 125.9] [#430148]

- After changing the time zone on a NetScaler appliance, you must restart the appliance so that policies referencing the LOCAL system use the new time zone instead of the old one. Otherwise, policies that should match do not, and policies that should not match do.

  [From Build 129.22] [#425465]

- The maximum value of the RelayState attribute that can be sent with the assertion that NetScaler sends is increased to 512 bytes. This applies to cases where the administrator configures a traffic policy to send assertion to a relying party.

  [From Build 129.22] [#473721]

- Using the "SYS.CHECK_LIMIT" expression in conjunction with any boolean expression can cause the NetScaler to crash.

  [From Build 129.22] [#493045]

- Rewrite policy bindings to virtual servers can be lost when you upgrade the NetScaler firmware to version 10.1.128.11. If the rewrite policy is bound to a load balancing virtual server, the policy bindings are not displayed as part of the server configuration, but they are saved when the user saves the configuration. If the rewrite policy is bound to a content switching virtual server, the policy bindings are lost when the user saves the configuration.

  [From Build 130.13] [#508510, 513724, 517150, 518535, 519945]

- The NetScaler appliance can crash or the data can get corrupted when the URL (or other string) satisfies the following criteria:

  - Length is more than 1300 bytes (800 bytes for HTML_XML_SAFE).

- Has at least one unsafe character.

- A significant initial part of the string does not need encoding (or some smaller initial part of the string does not need encoding and there are lots of characters needing encoding)

- One of the following functions is used on the string in the expression:

* HTTP_URL_SAFE - unsafe characters are not allowed. Safe characters are: a-z, A-Z, 0-9, "-", "_", ".", "!", "~", "*", "'", "(", ")", ";", ":", "@", "?", "=", "$", "%", "&", "+", ",", "/".

* HTTP_HEADER_SAFE - new line ('\\n') characters are unsafe.

* HTML_XML_SAFE - unsafe characters are '<', '>' and '&'.

* APPEND_QUERY_PARAMETER - same as HTTP_URL_SAFE

Workaround: As a workaround, remove uses of these functions from your expressions if strings can be long (or truncate the strings to 1300 bytes (800 bytes for HTML_XML_SAFE)). In a number of cases you can avoid using these functions if you concatenate the URL with some string constant to the left of it (for example "" + HTTP.REQ.URL) - if the input was encoded, so will be the result.

[From Build 130.13] [#506761, 519776]

- The default SSL virtual server configurations are disturbed, if HTTP callouts are configured on the NetScaler appliance.

  [From Build 132.8] [#551626]

- A NetScaler appliance that has a rewrite policy configured, becomes unresponsive, if all the following conditions are met:

  1. The rewrite action type is either "replace" or "insert_after".

  2. The HTTP response does not have the content-length header.

  3. The body of the HTTP response is split into multiple TCP packets with different TCP packets arriving with some time delay. This causes the policy rewrite engine to pause and resume the packet processing.

  4. The string specified in the rewrite action is present in the last packet of the HTTP response.

  [From Build 132.8] [#554460]

- Modifying the content with more than one callout results in incorrect computation of the content length. This issue is not observed if all the callouts use GET requests.

  [From Build 120.13] [#401455]

- On a NetScaler appliance with Rewrite enabled and configured, a newly-created Rewrite policy that is bound to a content-switching virtual server might not be saved either in the running configuration or in the saved configuration.

  [From Build 122.17] [#418252]

- A new SNMP alarm, vridStateChange, indicates the change of the state of a VRID from backup to master in an active-active configuration. The NetScaler appliance in which the state of a VRID changes to master sends a trap message for each VIP address bound to that VRID to the configured SNMP managers, indicating that the NetScaler appliance is currently serving traffic for a particular VIP address bound to that VRID. If no VIP addresses are bound to that VRID, the appliance does not send any trap messages.

  [From Build 118.7] [#246215]

- SNMPD fails to respond if it receives a packet with a NULL community string.

  [From Build 121.10] [#413733, 413871, 421055, 468830]

- Net-SNMP does not handle the endOfMibView condition properly if the value of max-repetition is set to zero. As a result, memory allocation failures, and the SNMP daemon fails to respond.

  [From Build 123.11] [#435520, 438590]

- The aggregateBWUseHigh and aggregateBWUseNormal SNMP traps are frequently generated even though the bandwidth is less than the set value for the alarm.

  [From Build 125.9] [#407594]

- The NetScaler appliance sometimes fails when a TCP connection is closed from a SPDY client while some streams are still active.

  [From Build 122.17] [#406948, 405903, 429211, 432515]

- Next Protocol Negotiation (NPN) TLS extension cannot be explicitly enabled or disabled. It is automatically enabled when SPDY is enabled on a HTTP profile, and disabled when SPDY is disabled.

  [From Build 127.10] [#460918, 474003]

- In some cases, parsing an incorrectly formatted client certificate might take more than a few seconds. The delay can trigger the monitoring logic to terminate the process and restart the appliance.

  [From Build 118.7] [#392683, 257157, 392686, 392996]

- An attempt to establish an HTTPS connection to a NetScaler FIPS appliance through a Chrome browser fails, because the browser sends a SPDY-NPN extension by default, and the NetScaler FIPS appliance does not support the NPN extension.

  Workaround: Disable SPDY in the Chrome browser.

  [From Build 119.7] [#400084]

- In the NetScaler configuration utility, the "FipsKey" parameter does not appear in the "Install certificate" dialog box. As a result, you cannot add a certificate-key pair on an MPX FIPS appliance by using the configuration utility.

  Workaround: Use the command line interface.

[From Build 119.7] [#400649]

- If any entity is added as part of user interactive process on command line for SSL Certificates and the operation is aborted in between using CTRL+C, then again carrying out the same operation causes the NetScaler command line to crash.

  [From Build 121.10] [#408393]

- If a malformed packet is received from a client, the NetScaler appliance closes the connection and releases the resources used for that connection to the common pool. In some cases, some of these resources are not cleaned before returning to the pool and a bad resource might be reused for a future request. In such cases, the SSL handshake for that future request fails.

  [From Build 122.17] [#423905, 418100, 430942]

- If a client sends a certain type of malformed message, which can make uninitialized resources available for subsequent handshakes, an SSL handshake that uses one of those resources causes a memory leak.

  [From Build 123.11] [#431919]

- If you create a certificate revocation list (CRL), enable refresh, and specify the method as HTTP or LDAP, CRL refresh does not happen.

  [From Build 123.11] [#434737]

- If you upgrade to this build, the number of SSL chips for which the status is shown as UP on an MPX 21550 platform with 36 chips is less than the actual number of chips that are UP. This is only a reporting issue.

  [From Build 123.11] [#235990]

- If the SSL handshake uses the TLSv1.1 or TLSv1.2 protocol and you have bound an RC4 cipher to the SSL virtual server, downloading a large file might take an unusually long time.

  [From Build 123.11] [#432375]

- On the NetScaler appliance, TLS protocol version 1.2 does not support ephemeral Diffie-Hellman cipher suites.

  [From Build 124.13] [#345883]

- If you add a certificate revocation list (CRL) with refresh enabled, the appliance might perform a core dump and restart.

  [From Build 125.9] [#436205, 411462, 436185]

- On a Nitrox-2 chip based platform, if you bind cipher groups, such as HIGH and AES, to your virtual server, the unsupported ECDHE cipher might also be bound. This cipher does not cause any problems. To remove it, you must unbind the cipher group.

  [From Build 126.12] [#437018]

- In a high availability setup, the force ha sync command appends the DEFAULT cipher group to the user-defined ciphers on the virtual server of the secondary node.

  [From Build 126.12] [#451698, 446674, 452080]

- If you use the configuration utility to configure FIPS appliances in a high availability setup, FIPS keys are not exported or imported between the nodes, because the option to enable secure information management (SIM) is not available.

  [From Build 127.10] [#459688, 446760]

- The version displayed in syslog is SSLv2.0 even though the session is negotiated using TLSv1.2.

  [From Build 128.8] [#474417, 474413]

- In rare cases, if the random number generated for the DH key exchange has a leading zero, DH negotiation fails because of a hardware limitation.

  [From Build 128.8] [#414388, 345883, 349858, 428257, 428259]

- On all the NetScaler MPX platforms, DH cryptographic operation is now offloaded to the hardware, reducing the load on the CPU. If your deployment uses DH crypto operations heavily, you will notice a performance improvement.

  [From Build 129.22] [#490273, 378182, 404081]

- If a spike in traffic occurs while the NetScaler ADC is doing a DH-based handshake, some packets might be dropped, because a DH handshake consumes a high number of CPU cycles.

  [From Build 129.22] [#484525]

- In a setup with a large number of virtual servers, if only a few virtual servers receive most of the traffic while the other virtual servers are idle, there might be a delay in cleaning up the sessions.

  [From Build 130.13] [#492087, 510038, 510483]

- If session reuse is enabled on the NetScaler and a network error occurs, the NetScaler attempts to clear the session information so that it is not reused for a subsequent session request from the same client. In rare cases, the NetScaler might fail during this cleanup process.

  [From Build 130.13] [#494093, 485932, 492191, 492797, 497321]

- On all NetScaler appliances except MPX 5500 and MPX 5550/5650/5750 appliances, if both the rate of new SSL connections and the percentage of SSL session reuse are high, SSL session buildup causes high usage of memory. If the result is a memory allocation failure, SSL traffic is dropped.

  [From Build 132.8] [#532136, 525686, 531207, 539902, 547350, 548697, 559753, 561598, 563485, 569063]

- If you run the "update ssl certkey" command to modify the certificate-key pair that is bound to a service group, a duplicate entry is seen for the same certificate key pair in the running configuration.

  [From Build 132.8] [#550138, 552436, 552701]

- SureConnect (SC) should be enabled on one entity. If you enable SC or configure SC policies on a load balancing virtual server, do not enable SC on any of the services or service groups that are bound to this virtual server. Doing so can result in configuration loss during reboot or lead to inconsistent configuration across an HA pair.

  [From Build 132.8] [#526782]

- When selective acknowledgement (SACK) and partial buffering are enabled on the appliance, acknowledgements with incorrect TCP checksum are forwarded to the server.

  [From Build 118.7] [#384153]

- The NetScaler appliance wrongly advertises TCP buffer size to the client side when dynamic windows management is enabled and the service-side buffer size is larger than 40k. This problem occurs when two different TCP profiles are bound to the virtual server (buffer size is 8k) and to the service (buffer size > 40k). It causes failure when the appliance is uploading files.

  [From Build 118.7] [#392293]

- SNMP returns incorrect values for the "ifOutOctets" and "ifInOctets" counters.

  [From Build 119.7] [#390257]

- If the SNMP service has the NSI_NS_SERVICE flag set, and you clear the configuration, the NetScaler appliance crashes.

  [From Build 119.7] [#404094]

- The SNMP module allocates memory for all OIDs in an SNMP request and queues them for further processing. With a large number of SNMP requests (each request with possibly hundreds of OIDs), the result can be a memory shortage that in turn leads to memory allocation failures.

  [From Build 119.7] [#394724, 411601]

- The NetScaler appliance dumps a core when you create a cluster or a high availability setup on an appliance that has a TFTP load balancing virtual server.

  Workaround: Make sure you delete existing TFTP load balancing virtual servers before creating the cluster or high availability setup.

  [From Build 119.7] [#395735, 401437, 406759, 407288]

- Stat-command output specified with the "fullValues" parameter is aligned incorrectly.

  [From Build 120.13] [#391632]

- When you try to add a second name-based SNMP manager, you get an error message that says an SNMP manger with that name already exists.

  [From Build 120.13] [#353546]

- If TCP buffering or caching is enabled on a NetScaler appliance receiving an ACK packet that has ACK_NO at the left edge of the SACK block, the packet engine enters a loop while processing the packet.

  [From Build 120.13] [#401111, 375425, 399769, 408267, 408648, 413721, 414273, 424726, 460731]

- Remote monitoring of a high capacity appliance, such as a NetScaler MPX 22000, might indicate a drop in performance even though performance remains robust. The apparent problem is the result of a pause in the stream of monitoring data, not an actual drop in throughput.

[From Build 120.13] [#407868]

- On a NetScaler MPX system, the SNMP count for the appliance's hardware memory and the show system memory display are incorrect. The amount of memory shown is larger than the actual amount.

  [From Build 120.13] [#391754]

- The NetScaler appliance might fail to respond if an ICMP error occurs when TCP buffering and integrated caching are enabled on the appliance.

  [From Build 120.13] [#402677, 406353, 408800, 411332, 412960, 426506, 441788]

- A session is not freed when port allocation fails. The session is getting matched and the NetScaler fails when it tries to access other linked sessions which are NULL.

  [From Build 120.13] [#407974, 421716]

- If, from a management computer, you run a command that forms a request size of more than 8000 bytes, the NetScaler ADC might not properly buffer this large request. As a result, the ADC terminates the connection to the management computer.

  [From Build 120.13] [#423610, 436854]

- The NetScaler appliance can crash when there are split ICA frames that span 2 CGP frames with other CGP packets in between.

  [From Build 121.10] [#411613, 414137, 436849, 444308]

- If changes are made in the nsconfig/resolv.conf file, the appliance fails to override the default DNS configurations.

  [From Build 121.10] [#412681]

- If you specify an invalid IPv4 address in a command that can accept either IPv4 or IPv6 address, the NetScaler shell exits automatically, because of to memory corruption.

  [From Build 121.10] [#415623, 247585, 327131, 384988]

- In an high availability setup, after a forced failover, the sync operation fails to sync the -establishClientConnection parameter setting.

  [From Build 121.10] [#216272, 220771]

- On a NetScaler appliance, an invalid HTTP range request results in a large amount of memory usage and the following error appears: "ERROR: Communication error with the packet engine."

  [From Build 121.10] [#401526]

- If TCP buffering or caching is enabled on a NetScaler appliance receiving an ACK packet that has ACK_NO at the left edge of the SACK block, the packet engine enters a loop while processing the packet.

  [From Build 121.10] [#375425]

- TCP buffering bypasses as the calculated 'usable system memory' is less than the configured threshold value.

[From Build 121.10] [#405532, 423815, 434383]

- The NetScaler appliance stops sending TCP DUP ACKs when it receives out of order packets. This might result in latency between the client and the appliance, or the appliance and the server, with reduced throughput for some traffic patterns.

  [From Build 122.17] [#417793, 421214, 421329, 423099]

- The NetScaler appliance does not forward the complete request to the server if the request requires more than one packet. As a result, the transaction fails.

  [From Build 122.17] [#420781]

- The NetScaler appliance intermittently resets TCP connections that originate from the NetScaler FreeBSD shell and are destined for NetScaler-owned IP addresses (for example, a SNIP or VIP address). The resets affect applications such as LDAP.

  [From Build 122.17] [#430176, 430185]

- When the NetScaler appliance receives invalid Selective Acknowledgment (SACK) blocks from the client, it attempts to send old data that has already been cleared. As a result, the appliance stops responding.

  [From Build 122.17] [#419553, 423433, 426506, 428155]

- When upgrading from release 9.3 to 10.1, the following SNMP alarms throw a time argument error: IP-CONFLICT, HA-LICENSE-MISMATCH, and HA-PROP-FAILURE. This issue occurs because, in version 10 and later, the "time" parameter is deprecated for these SNMP alarms.

  Note: The same error occurs if you try to set the time for one of these alarms.

  Workaround: Before upgrading to release 10.1, update the "ns.conf" file by removing the "time" parameter for these three alarms from the "set snmp alarm" command.

  [From Build 123.11] [#388481, 391618]

- On the "System > Diagnostics" page, when you select "Saved v/s running", the configuration utility displays a difference between the running and saved configurations, even if there is no difference.

  [From Build 123.11] [#388836, 388830, 388831, 411627, 416264, 430646, 430652]

- The "stat system -detail" command does not display the number of CPUs.

  [From Build 123.11] [#382647]

- The NetScaler appliance might fail to respond if an ICMP error causes the packet engine to enter a loop and thereby resulting in a pitboss process failure.

  [From Build 124.13] [#436798, 438765, 439849, 449803]

- When Call Home is enabled, duplicate SNMP traps are generated for power supply unit (PSU) failures.

  [From Build 124.13] [#435796]

- The MPTCP data_ack signal is not sent in the subflow in which the MP_FAIL signal is sent.

[From Build 124.13] [#397587]

- ISIS packets are dropped at the Nexus 1000V distributed virtual switch (DVS), which has no option to enable promiscuous mode. However, this issue is not observed when the virtual machines are connected through the ESX virtual switch with promiscuous mode ON.

  [From Build 124.13] [#430071]

- A signed short integer overflow can occur during packet processing. Subsequent packets are corrupted.

  [From Build 124.13] [#432728]

- If large number of small packets are sent through the packet processing pipeline, the packet engine enters a loop and restarts, causing a pitboss failure.

  [From Build 124.13] [#439579, 442723, 442749]

- When a client's MPTCP token is invalid in the C2C steered MP_CAPABLE final ACK, the packet is dropped silently without flushing out the RSS filter. This filter is never deleted. If the client reuses the same 4-tuple as the filter, the incoming packet may go into the steering loop between the PEs. This will lead to very high CPU utilization.

  [From Build 125.9] [#447623]

- When web server logging and audit logging are enabled on the NetScaler, the TCP current clients counter goes to negative values and shows a very large value in the stat or the SNMP OID.

  [From Build 126.12] [#335202, 248103, 341155, 404099]

- The NetScaler ADC forwards unprocessed packets to the load balancing virtual servers without selecting a service, because of an HTTP out-of-order packet processing issue. Instead of being dropped, these connections queue up at the virtual servers. The ADC fails to respond while processing these connections.

  [From Build 126.12] [#432612, 426784, 434780, 468253]

- The state of services for which NATPCB is allocated starts flapping because of NATPCB allocation failure.

  [From Build 126.12] [#453811, 470299]

- Memory leak found in shell '/bin/sh' while performing management CPU profiling in "nsproflog.sh" thereby causing swap zone issues.

  [From Build 126.12] [#462797, 441758, 446780, 455911, 457505, 459435, 468798, 476812, 495481]

- High CPU usage is observed when evaluating listen policy named expressions on a virtual server that picks up every packet.

  [From Build 126.12] [#450580]

- The NetScaler ADC might fail during an nstrace operation.

  [From Build 126.12] [#446300]

- The NetScaler appliance drops a connection if it receives 255 back-to-back old packets (re-transmissions). The limit is configurable and the default value has been increased.

[From Build 126.12] [#453108]

- If TCP buffering or caching is enabled on a NetScaler appliance receiving an ACK packet that has ACK_NO at the left edge of the SACK block, the packet engine enters a loop while processing the packet.

  [From Build 126.12] [#451285, 441843, 457850]

- With large number of configuration entries in the ns.conf file, the commands in the /nsconfig/rc.netscaler file might not be applied after the appliance is restarted.

  [From Build 126.12] [#396628, 402205]

- The "show ns runningConfig" command may produce partial output if invoked while another "show ns runningConfig" command, from the same or other admin session is in progress. **Workaround:** Re-execute the "show ns runningConfig" command to fetch the entire running configuration.

  [From Build 127.10] [#478895]

- When the NetScaler has application firewall disabled but SSO enabled, and if the NetScaler memory is less, all unused memory (appfw memory) is not recovered. This leads to an erroneous value for the "ActualInUse" memory counter.

  [From Build 127.10] [#450054, 450787, 453207, 453481, 459354]

- The NetScaler system backup tar file does not include the following files:

  - /nsconfig/ns.conf

  - /nsconfig/Zebos.conf

  - /nsconfig/rc.netscaler

  - /nsconfig/snmpd.conf

  - /var/log/wicmd.log

  - /nsconfig/nsbefore.sh

  - /nsconfig/nsafter.sh

  [From Build 127.10] [#455041, 478635, 484981]

- The NetScaler nstrace utility does not filter out all IPv6 packets when a IPv4 only filter is entered.

  [From Build 127.10] [#450398]

- The Monupload process monitors the power supply and sends a "show techsupport" bundle as soon as a power failure is observed. This behavior is now modified to upload the bundle only in case the power supply does not recover in a 1 minute.

  [From Build 128.8] [#452240]

- When different TCP profiles are bound to a virtual server and to the services that are bound to that virtual server, and one of the profiles has window scaling as ENABLED and the other has it as DISABLED, NetScaler sometimes considers that window scaling is ENABLED. The expectation in such a case is that NetScaler considers window scaling as

DISABLED.

[From Build 128.8] [#481442]

- Changes made to the time zone are not reflected till the NetScaler appliance is warm rebooted.

  [From Build 129.22] [#471100, 425465, 484159, 484187]

- With USIP mode enabled, when the client FIN comes along with the final ACK for the server response, the NetScaler TCP module does not acknowledge the FIN.

  [From Build 129.22] [#478356]

- A new HTTP profile option "rtspTunnel" allows RTSP over HTTP. The RTSP tunnel is detected by the presence of either one of the following

  - 'Accept: application/x-rtsp-tunnelled' request header

  - 'Content-Type: application/x-rtsp-tunnelled' response header

  Once the tunnel is detected, NetScaler stops HTTP tracking for that TCP connection and lets the RTSP flow go through. The "rtspTunnel" option is disabled by default.

  [From Build 129.22] [#480219]

- When the Call Home feature is disabled before the Call Home enable operation is successful, a second instance of the Call Home process starts to run. This results in high usage of the management CPU.

  [From Build 129.22] [#498232]

- The NetScaler intermittently fails to generate traps due to issues in propagating the alarm state to the SNMP daemon.

  [From Build 129.22] [#490192]

- If you change the IP address of a load balancing virtual server that shares the same server information (IP address, port and service) with an audit server and then clear the configurations, the NetScaler is expected to remove the virtual server, the audit server, and other NetScaler configurations. However, when you now add the virtual server with the original server details, the NetScaler throws an error message that says "resource already exists".

  Note: In a HA setup, this behavior is displayed even when you perform a force sync or a force failover operation.

  [From Build 129.22] [#484527]

- SNMP walk shows the operational status of a LA channel as DOWN even when it is in the PARTIAL-UP state.

  [From Build 129.22] [#477709]

- The nsnetsvc process size increases when the "stat" command is executed.

  [From Build 130.13] [#418028, 409722, 467187]

- When the Netscaler ADC hits congestion with HA or LACP packets or continuous congestion in a single-PE environment, it cannot recover and packet transmission stops. This is applicable to the management ports on NetScaler SDX appliances and to all ports on NetScaler VPX instances running on XenServer.

[From Build 130.13] [#532316, 532045, 533018, 534634, 534671, 537616]

- The NetScaler appliance can crash when a large HTTP request URL has a space in it and if the request is broken into multiple packets.

  [From Build 130.13] [#497321, 501856, 502116, 502902, 517374]

- When a HTTP profile is bound to a virtual server or service, the configurations of this profile are considered over the configurations of the global HTTP profile (nshttp_default_profile). However, when connection multiplexing is disabled globally and enabled on the virtual server or service, the global setting for connection multiplexing is being considered. This issue has now been fixed.

  [From Build 130.13] [#494013]

- The NetScaler randomly crashes when SPDY is enabled on a NetScaler deployment which has integrated caching or front end optimization enabled. This occurs due to some interaction issues.

  Workaround: Disable SDPY when integrated caching or front end optimization is enabled.

  [From Build 131.11] [#486257]

- If you enable the nstrace feature in TX mode with an advanced filter expression, the NetScaler appliance fails.

  [From Build 131.11] [#494911, 481032, 511763, 528309, 532708, 538507]

- If password based authentication is used to open an SSH session to a NetScaler appliance, the wrong remote IP address is sent to the NetScaler syslog records.

  [From Build 131.11] [#286861, 301935, 513312, 522183, 541332]

- The ns_monuploadd_err.pl script monitors the health of the NetScaler appliance by looking for errors recorded in the log files. The script decompresses the log files and does not remove the decompressed log files, which therefore consume disk space.

  [From Build 131.11] [#532042, 447664, 532587, 533164]

- A NetScaler ADC processing SPDY traffic on SPDY enabled virtual servers fails intermittently if an HTTP response body received with chunked transfer-encoding and the response header is modified by other NetScaler features.

  [From Build 131.11] [#519004, 528861]

- If a non-HTTP request is received on an HTTP virtual server, the transaction might fail.

  [From Build 131.11] [#504910]

- When upgrading the NetScaler software from release 9.3, without a cache license, to release 10.0 or later, with a cache license, you have to apply the cache configuration manually to enable the integrated caching feature.

  [From Build 131.11] [#451841, 332826, 346327, 361979, 465489, 485864]

- The save ns config command and the nsnetsvc process fail under low memory conditions.

  [From Build 131.11] [#488110, 496136]

- If you enable SPDY and the SPDY layer accumulates more than 8912 bytes of set-cookie values while processing a sever response, a buffer overrun causes the NetScaler appliance to fail.

  [From Build 131.11] [#524949]

- The NetScaler backup and restore functionality now creates a backup of each of the following configuration files: inetd.conf, ntp.conf, syslog.conf, newsyslog.conf, crontab, host.conf, hosts, ttys, sshd_config, httpd.conf, monitrc, rc.conf, ssh_config, localtime, issue, and issue.net.

  [From Build 131.11] [#506378]

- A NetScaler VPX virtual appliance with multiple packet engines fails if you enable the nstrace feature in TX mode with an advanced filter expression.

  [From Build 131.11] [#528309]

- If the NetScaler appliance uses the HTTP pipeline to parse an HTTP request, and the parsing process fragments the request packet, the appliance might not UNSET the NS_FINAL_DATA flag after receiving a fragment of the packet. In that case, the appliance will fails.

  [From Build 131.11] [#527320, 527211]

- During the execution of the "nstrace.sh" script (from shell) or the "start nstrace" command (from CLI), when the trace file is rolled over, some packets might not be available in the trace. The number of packets that will be dropped from the trace is directly proportional to the traffic rate.

  [From Build 132.8] [#480258, 494482, 523853]

- Enabling the AppFlow feature during a transaction causes the NetScaler appliance to fail.

  [From Build 132.8] [#547739, 527797, 531101]

- The output of the "show channel" command includes interfaces that have been unbound from the channel.

  [From Build 132.8] [#540998]

- If the NetScaler appliance receives a Websocket upgrade request, and an HTTP-body based policy is bound globally or to a virtual server, the appliance does not forward the request to server until a TCP FIN flag is received from the client.

  [From Build 132.8] [#536576, 549318]

- In a cluster setup, if the TCP profile parameter 'sendBuffsize' is unset the NetScaler appliance displays 0 bytes as the buffer size instead of 8190 bytes (default value).

  [From Build 132.8] [#552654]

- Multiple instances of the nstraceaggregator daemon can run at the same time. As a result the NetScaler appliance might fail and corrupt the captured files.

  [From Build 132.8] [#527119, 522584, 525657]

- A NetScaler appliance fails if it attempts to apply HTML injection to a server response that does not have a content type header.

[From Build 132.8] [#529493]

- The memory allocation API, malloc, returns a NULL value if it does not obtain memory for the nscollect utility. If the nscollect utility tries to dereference this NULL pointer, the result is a memory segmentation error.

  [From Build 131.11] [#528818, 529425]

- The NetScaler ADC generates SNMP clear alarm traps for successful cases of haVersionMismatch, haNoHeartbeats, haBadSecState, haSyncFailure, and haPropFailure error events in an HA configuration.

  [From Build 131.11] [#368832]

- The SNMP counter of type cntr32 has been changed to a gauge counter.

  [From Build 131.11] [#524080, 448724]

- In certain cases, an attempt to add or bind a load balancing virtual server, service, or service group can fail if the internal ID assigned to the virtual server, service or service group conflicts with the internal ID of an existing virtual server, service, or service group.

  Workaround: Try creating the virtual server, service, or service group again.

  [From Build 132.8] [#516162, 358664, 538009, 540912, 542248, 542721, 546566, 549368]

- The NetScaler appliance serves erroneous cache content if you use the XenApp/XenDesktop wizard's auto-configured cache policies.

  [From Build 132.8] [#426551, 545422]

- You can now optionally configure agCallbackURL from agURL. The agURL would represent the front end Access Gateway (AG) for the client. The agCallback is for communication between Web Interface (WI) and AG. Also, The agCallbackURL is an optional parameter. Use the following command to configure agCallbackURL:

  add wi site /Citrix/new http://agee.citrix.com http://sta.citrix.com -agCallbackUrl http://callback.citrix.com

  [From Build 131.11] [#508743]

- Upgrading a NetScaler ADC from release 10 to release 10.1 deletes a set of customized options of the add wi site command.

  [From Build 126.12] [#456120]

- In a high availability setup, if the failover operation is performed twice, a user trying to launch an application is unable to proceed after the AGESSO.jsp page appears. If the domain controller is configured for x number of logon retries, and the user refreshes the page x number of times, the account is locked.

  With this fix, the user is able to launch the application. However, if an application is launched immediately after failover, and the launch takes longer than usual (about 75 seconds), a session error page might appear, in which case the user has

to log on again.

[From Build 126.12] [#450811]

- Neither the CLI nor the configuration utility allows a user to configure a pre-login message of more than 255 characters.

  [From Build 126.12] [#458113]

- Users who access a Microsoft Sharepoint server through a NetScaler ADC that has the application firewall enabled are unable to open any document type that requires software that is not part of the browser, such as Microsoft Office files.

  [From Build 129.22] [#450232]

Enhancement Releases

Apr 08, 2014

This section describes the enhancements and known issues provided in the enhancement releases of the Citrix NetScaler and Citrix NetScaler SDX.

- Build 129.1105.e
- Build 127.1007.e
- Build 126.1203.e
- Build 124.1308.e
- Build 123.1100.e
- Build 122.1708.e
- Build 121.1013.e

Build 129.1105.e

Feb 13, 2014

Release version: Citrix NetScaler release 10.1.e build 129.1105.e

Replaces build: None

Release date: Oct 2014

Release notes version: 1.0

Language supported: English (US)

Review the following sections:

Note:

- This release is based on Citrix NetScaler release 10.1 build 129.11. The release notes are available in the Build 129.22 section on Citrix eDocs.
- The enhancements and known issues in this release apply to Citrix NetScaler 10.1.e nCore.

Bug Fixes

May 07, 2014

Networking

- Issue ID 490341: With MAC based forwarding (MBF) option enabled, the NetScaler ADC does not update Layer 2 information such as MAC address, interface ID, and VLAN ID, for a dynamic service even when the associated router is inactive. As a result, the router drops the packets destined to the IP address specified by the dynamic service.

Build 127.1007.e

Feb 13, 2014

Release version: Citrix NetScaler release 10.1.e build 127.1007.e

Replaces build: None

Release date: Aug 2014

Release notes version: 1.0

Language supported: English (US)

Review the following sections:

Note:

- This release is based on Citrix NetScaler release 10.1 build 127.10. The release notes are available in the Build 127.10 section on Citrix eDocs.
- The enhancements and known issues in this release apply to Citrix NetScaler 10.1.e nCore.

Bug Fixes

May 07, 2014

High Availability

- Issue ID 469857: On a HA setup, even though the source IP is not explicitly set to *, the output of the "show ns rpcNode" commands shows the source IP as *. Therefore, when HA failover happens for the second time, the LB persistency session information is not propagated to the secondary node. This means that the information is not available when a forced failover is performed on the new primary node.
  The fix ensures that the NetScaler IP (NSIP) address of the local box is always set as the source IP address in a HA setup.

NetScaler Gateway

- Issue ID 484245: If Kerberos uses x.509 certificates (PKINIT) for single sign-on, NetScaler Gateway fails to obtain tickets if the Key Distribution Center (KDC) returns a realm referral. This can cause the NetScaler Gateway appliance to fail.
- Issue ID 461279: When users upgrade the NetScaler Gateway Plug-in from Version 10.1.122.17 or later to the latest Version 10.1 Maintenance Release on a computer that includes an installation of Citrix Receiver, the automatic upgrade fails.
- Issue ID 463871: If you bind SAML and LDAP authentication polices to the virtual server for two-factor authentication, after authenticating with SAML which is primary authentication type the LDAP user name populates automatically. If the first logon attempt to LDAP fails, user names are case-sensitive and must be entered again exactly as it appears after SAML authentication. For example, if the user name is populated as JohnDoe@xyzz.com and the user types johndoe@xyzz.com during the subsequent attempt, log on fails.
- Issue IDs 481889, 486176: In a high availability deployment, if the NetScaler Gateway virtual server is missing on the secondary appliance, NetScaler Gateway fails during session propagation.

NetScaler SDX Appliance

- Issue ID 423917: DNS configuration is not included in backup files
  Workaround:Configure the DNS configuration through network settings option

Build 126.1203.e

Feb 13, 2014

Release version: Citrix NetScaler release 10.1.e build 126.1203.e

Replaces build: None

Release date: June 2014

Release notes version: 1.0

Language supported: English (US)

Review the following sections:

- Enhancements

Note:

- This release is based on Citrix NetScaler release 10.1 build 126.12. The release notes are available in the Build 126.12 section on Citrix eDocs.
- The enhancements and known issues in this release apply to Citrix NetScaler 10.1.e nCore.

Enhancements

May 07, 2014

- Issue ID 437359: A parameter Source IP Persistency has been introduced in RNAT rules and Netprofiles:
  Source IP Persistency for RNAT Sessions

  The source IP persistency of a RNAT rule enables the NetScaler ADC to use the same NAT IP address for all RNAT sessions initiated from a particular server.

  Source IP Persistency for NetProfiles

  The source IP persistency of a netprofile associated with a virtual server or service enables the NetScaler ADC to use the same address, specified in the net profile, for all sessions initiated from a particular client.

Build 124.1308.e

Apr 08, 2014

Release version: Citrix NetScaler release 10.1.e build 124.1308.e

Replaces build: None

Release date: April 2014

Release notes version: 1.0

Language supported: English (US)

Review the following sections:

- Enhancements

Note:

- This release is based on Citrix NetScaler release 10.1 build 124.13. The release notes are available in the Build 124.13 section on Citrix eDocs.
- The enhancement in this release apply to Citrix NetScaler 10.1.e nCore.

Enhancements

May 07, 2014

- ENH ID 0413833: You can now use Remote Integrated Service Engine (RISE) technology to integrate a NetScaler ADC and a Cisco Nexus 7000 Series switch. This combination offers layered network services, including robust application delivery capabilities that accelerate application performance for all users.
  With a RISE based implementation, the NetScaler functionality is available as a centralized resource that can be leveraged across the application infrastructure supported by the Cisco Nexus 7000 series switch. The key functionalities of the RISE architecture include:

  - Plug and play auto-provisioning
    RISE provides a plug and play auto-provisioning feature. When you directly connect the NetScaler ADC to the Cisco Nexus 7000 series switch, auto-discovery commences.

  - Discovery and bootstrapping
    The discovery and bootstrap mechanism enables the Cisco Nexus 7000 Series switch to communicate with the NetScaler ADC by exchanging information to set up a RISE channel, which transmits control and data packets.

  - Health Monitoring
    The NetScaler ADC uses its health monitoring feature to track and support server health by sending health probes to verify server responses.

  - Automatic Policy Based Routing (APBR)
    Automatic Policy Based Routing (APBR) automatically routes the return traffic from the servers to the NetScaler ADC, preserving the client IP addresses. The automatic policy based routes are defined on the Cisco Nexus 7000 series switch. When the return traffic from the server reaches the Cisco Nexus 7000 series switch, the APBR policies defined on the switch route the traffic to the NetScaler ADC, which in turn routes the traffic to the client.

- ENH ID 0436500: In a High Availability (HA) setup, stateful connection failover is now supported for load balancing virtual servers configured in TOS mode.
  With stateful connection failover enabled, the secondary appliance has information about the connections established before the failover and starts serving those already established connections after the failover.

  After HA failover, the client remains connected to the same physical server. The new primary appliance synchronizes the information with the new secondary appliance by using the SSF framework. During the transition period, the client and server may experience a brief disruption and retransmissions.

Build 123.1100.e

Mar 20, 2014

Release version: Citrix NetScaler release 10.1.e build 123.1100.e

Replaces build: None

Release date: March 2013

Release notes version: 2.0

Language supported: English (US)

Review the following sections:

- Enhancements
- Known Issue

Note:

- This release is based on Citrix NetScaler release 10.1 build 123.11. The release notes are available in the Build 123.11 section on Citrix eDocs.
- The enhancements and known issue in this release apply to Citrix NetScaler 10.1.e nCore™.

Enhancements

Mar 20, 2014

- ENH ID 0368447: This enhancement allows state information, in the form of variables, to be stored and used on NetScaler appliances . Variables can be of ulong, text, or map types. A map can have ulong and text type elements. And the map key is always text.

  Note:
  - Variables are not supported in a high availability setup.
  - Once configured, a variable's settings cannot be modified or reset. If the variable needs to be changed, the variable and all references to the variable (expressions and assignments) need to be deleted. Then the variable can be re-added with new settings, and the references (expressions and assignments) can be re-added.

  To use variables by using the command line interface

  1. Create a variable. Variables can be of singleton (ulong and text) and map type.
     //Declares a single valued 64-bit integer variable named my_counter. It is initialized to 0.
     ```
     add ns variable my_counter –type ulong
     ```
     //Declares a map named user_privilege_map that will contain keys of maximum length 15 characters and text values of maximum length 10 characters, with a maximum of 10000 entries. If the map contains 10000 unexpired entries, assignments for new keys reuse one of the least recently used entries . By default, an expression trying to get a value for a non-existent key will initialize an empty text value.
     ```
     add ns variable user_privilege_map -type map(text(15),text(10),10000)
     ```
  2. Specify the assignment for the variable. The assignment specifies the value or operation to be performed on that variable.
     //Defines an assignment named inc_my_counter that automatically adds one to the my_counter variable.
     ```
     add ns assignment inc_my_counter -var $my_counter -add 1
     ```
     //Defines an assignment named set_user_privilege that adds to the user_privilege_map variable an entry for the client's IP address with the value returned by the get_user_privilege HTTP callout. If an entry for that key already exists, the value will be replaced. Otherwise a new entry for the key and value will be added. Based on the previous declaration for user_privilege_map, if the map already has 10000 entries, one of the least recently used entries will be reused for the new key and value.
     ```
     add ns assignment set_user_privilege -var $user_privilege_map[client.ip.src.typecast_text_t] -set sys.http.callout(get_user_privilege)
     ```
     //Defines an assignment named clear_user_privilege that clears the entry for the client's IP address in the user_privilege_map variable.
     ```
     add ns assignment clear_user_privilege -var $user_privilege_map[client.ip.src.typecast_text_t] -clear
     ```
  3. Configure the assignment as an action for a policy.
     //Configures the assignment set_user_privilege with a compression policy
     ```
     add cmp policy set_user_privilege_pol -rule $user_privilege_map.valueExists(client.ip.src.typecast_text_t).not -resAction set_user_privilege
     ```

  To use variables by using the configuration utility

  1. Navigate to AppExpert > NS Variables, to create the variables.
  2. Navigate to AppExpert > NS Assignments, to assign values to the variables.
  3. Navigate to the appropriate feature area where you want to configure the assignment as an action.

For more information, see Variables.

Known Issue

Mar 12, 2014

- Issue ID 0419226: In the configuration utility, the online help for the content accelerator feature mentions a video that is not available.

Build 122.1708.e

Feb 13, 2014

Release version: Citrix NetScaler release 10.1.e build 122.1708.e

Replaces build: None

Release date: February 2014

Release notes version: 1.0

Language supported: English (US)

Review the following sections:

- Enhancements
- Known Issue

Note:

- This release is based on Citrix NetScaler release 10.1 build 122.17. The release notes are available in the Build 122.17 section on Citrix eDocs.
- The enhancements and known issues in this release apply to Citrix NetScaler 10.1.e nCore.

Enhancements

Feb 13, 2014

Adaptive Threshold in NetScaler Insight Center

- ENH ID 0378995: The adaptive threshold functionality in NetScaler Insight Center dynamically sets the threshold value for the maximum number of hits on each URL. If the maximum number of hits on a URL is greater than the threshold value set for the URL, a syslog message is sent to an external syslog server. The threshold value can be set for daily or weekly interval.

  For more information, see Managing Threshold.

Provisioning Palo Alto VM-Series Instances on a NetScaler SDX Appliance

- ENH ID 0357214: Palo Alto Networks VM-Series on Citrix NetScaler SDX enables consolidation of best-in-class security and ADC capabilities on a single platform, for secure, reliable access to applications by businesses, business units, and service-provider customers. The combination of VM-Series on Citrix NetScaler SDX also provides a complete, validated, security and ADC solution for Citrix XenApp and XenDesktop deployments.

  You can provision, monitor, manage, and troubleshoot an instance from the Management Service.

  Note: The total number of instances that you can provision on an SDX appliance depends on the license installed on the appliance.
  Important: You must upgrade your XenServer version to version 6.1.0 and install the xs-netscaler-6.1.0-2.6.32.43-0.4.1.xs1.6.10.777.170770-100012 supplemental pack.
  For more information, see Palo Alto Networks VM-Series.

Known Issue

Mar 12, 2014

Configuration Utility

- Issue ID 0419226: In the configuration utility, the online help for the content accelerator feature mentions a video that is not available.

Build 121.1013.e

Dec 31, 2013

Release version: Citrix NetScaler release 10.1.e build 121.1013.e

Replaces build: None

Release date: December 2013

Release notes version: 1.0

Language supported: English (US)

Review the following sections:

- Enhancements
- Known Issue

Note:

- This release is based on Citrix NetScaler release 10.1 build 121.10. The release notes are available in the Build 121.10 section on Citrix eDocs.
- The enhancements and known issue in this release apply to Citrix NetScaler 10.1.e nCore™.

Enhancements

Jan 01, 2014

Authentication and Authorization Enhancements

- ENH ID 0399086: With this release, the following authentication and authorization capabilities are supported on NetScaler SDX appliance:

  - External authentication for RADIUS, TACACS, and LDAP servers.
  - Group extraction capability for LDAP and RADIUS authentication types.
  - Authentication and authorization for requests through SSH. However, the authorization of SSH users is limited to super-user privileges only.
  - Audit logs for RADIUS and TACACS servers. You need to enable the Accounting option for the servers in the Management Service.

  For more information, see Configuring Authentication and Authorization Settings.

User Name and Password Length Extended to 127 Characters

- ENH ID 0325421: User names and passwords on the NetScaler appliance can now be up to 127 characters in length. Usernames and passwords can consist of upper-case and lower-case letters, digits, and the hyphen and underscore characters.

Content Accelerator

- ENH ID 0400961: The NetScaler provides a feature called Content Accelerator, that can be used in a Citrix ByteMobile T1100 deployment, to store content on a Citrix ByteMobile T2100 appliance. For more information, see Content Accelerator.

Known Issue

Jan 01, 2014

Configuration Utility

- Issue ID 0419226: In the configuration utility, the online help for the content accelerator feature mentions a video that is not available.

Configuration Utility Changes

Jul 11, 2013

In release 10.1, the NetScaler configuration utility has a new look. The navigation tree is reorganized and grouped according to the major features of the NetScaler appliance. The action buttons have been moved to the top of the screen, and additional actions are now in a drop-down list. Some of the subnodes are also moved to the right pane of the configuration utility.

Note: In eDocs, only topics that have been updated or added in the current release reflect the interface changes. For other topics, see the "Node Mapping table" below to map the former top-level nodes to their new locations.

Navigation Tree Reorganization

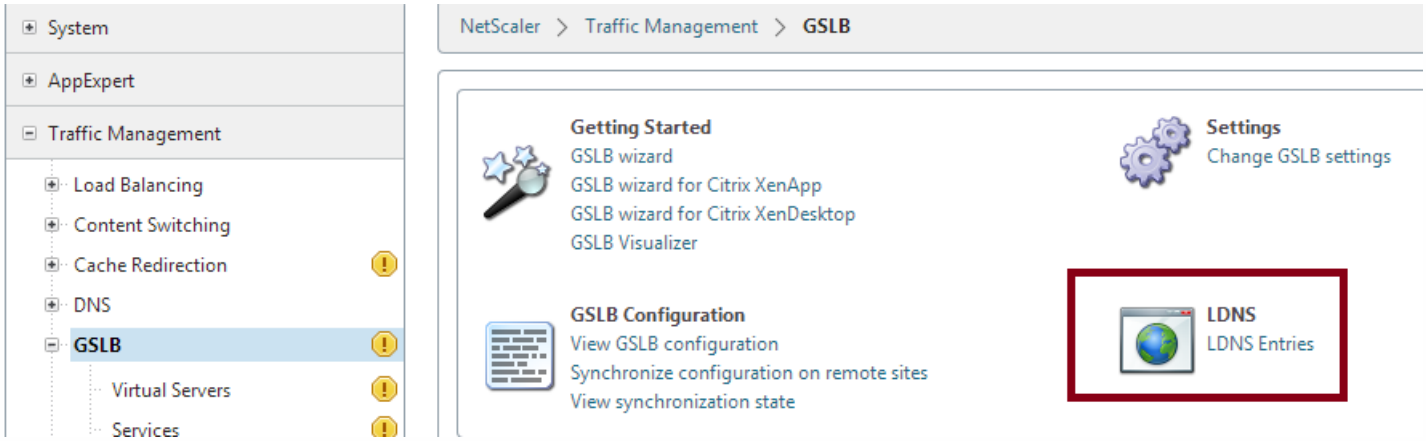
The nodes in the configuration utility are grouped by feature area, in the same way that feature documentation is organized in eDocs. The top-level nodes in the navigation tree of the configuration utility are **System**, **AppExpert**, **Traffic Management**, **Optimization** and **Security**. The following table shows the new location of nodes that were formerly at the top-level.

Table 1. Node mapping table

Top-level nodes in the previous version of configuration utility	Location of the nodes in current configuration utility
Network	System > Network
Group	System > User Administration
Users	System > User Administration
Database Users	System > User Administration
command policies	System > User Administration
EdgeSight Monitoring	System > EdgeSight Monitoring
Network	System > Network
CloudBridge	System > Cloud bridge
Web interface	System > Web Interface
Rewrite	AppExpert > Rewrite
Responder	AppExpert > Responder

Top-level nodes in the previous version of configuration utility	Location of the nodes in current configuration utility
Load Balancing	Traffic Management > Load balancing
Content Switching	Traffic Management > Content Switching
Cache Redirection	Traffic Management > Cache Redirection
DNS	Traffic Management > DNS
GSLB	Traffic Management > GSLB
SSL	Traffic Management > SSL
SSL Offload	Traffic Management > SSL Offload
HTTP Compression	Optimization > HTTP Compression
Integrated Caching	Optimization > Integrated Caching
AAA-Application traffic	Security > AAA-Application traffic
Application Firewall	Security > Application Firewall
Protection features	Security > Protection features
Other changes	
GSLB > Location	AppExpert > Location

In addition to the reorganization of the nodes within the navigation tree, some of the nodes are now grouped along with the configurations options present in the right pane of the configuration utility. For example, LDNS Entries which was earlier present as a sub-node of GSLB, is now grouped along with the global GSLB configurations present in the details pane of the configuration utility.



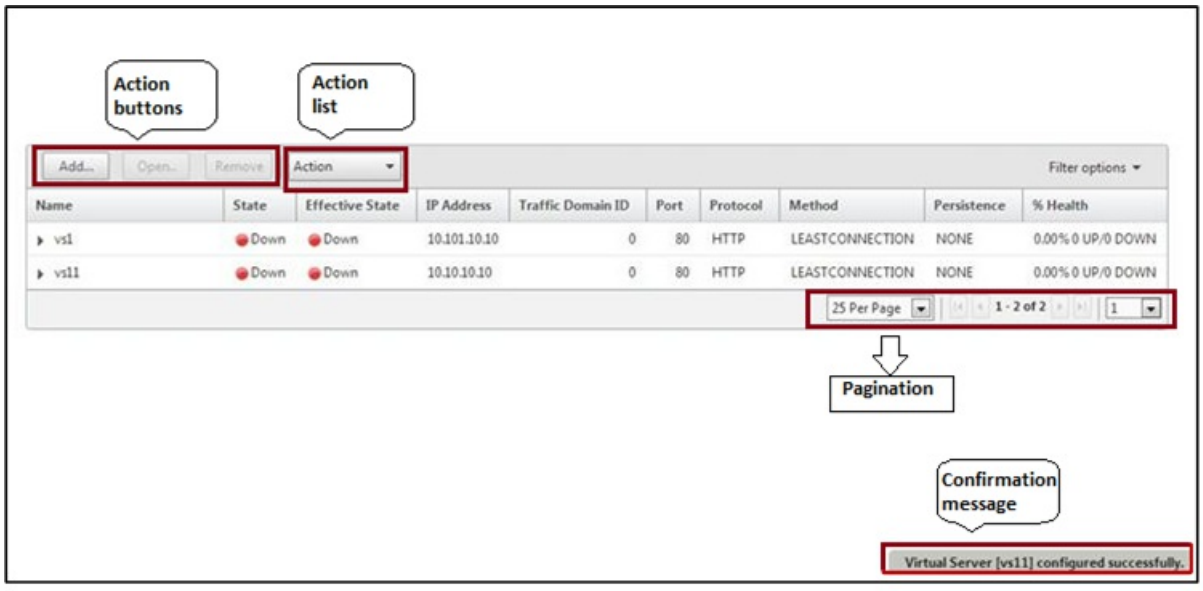
- Auto Detected Services, earlier present as a separate tab when you select Load Balancing > Services is now present as an option in the **Action** list when you select Load Balancing > Services.
- The option to view FIPS Configuration Summary is now present in the details pane, when you select **SSL**.
- The options to view the AppExpert applications, application templates, and the NetScaler Gateway Applications is present in the details pane, when you select **AppExpert**.
- The option to view cache objects is now present in the details pane, when you select **Integrated Caching**.

To locate any node in the new configuration utility, see the *Node mapping* table.

Other Changes

The following changes further enhance the usability of the configuration utility:

- The (Add, Open, and Remove) action buttons are now at the top of the details pane.
- The other action buttons (for example, Rename, Show Bindings, Policy Bindings, Statistics) have been replaced by Action list.
- The option for number of records to display per page has been moved to the bottom of the table that lists the entities.
- The confirmation message for any action that the user performs is displayed at the lower right corner of the configuration utility.
- A vertical scroll bar is provided in the right pane of the configuration utility for easy viewing.



NetScaler Licensing Overview

Aug 24, 2016

If you want to upgrade your software to 10.1, you can use your existing license. Contact your Citrix sales representative for new licenses if you are using the standard edition and want to upgrade to the enterprise or platinum edition, or if you are using the enterprise edition and want to upgrade to the platinum edition.

You can easily allocate your NetScaler licenses. In the NetScaler configuration utility (GUI), you can use your hardware serial number (HSN) or your license activation code (LAC) to allocate your licenses. Alternatively, if a license is already present on your local computer, you can upload it to the appliance.

For all other functionality, such as returning or reallocating your license, you must use the licensing portal. Optionally, you can still use the licensing portal for license allocation. For more information about the licensing portal, see "http://support.citrix.com/article/CTX131110".

Note:

- On a NetScaler MPX or SDX appliance, you can use the HSN or LAC to allocate your license or upload the license to the appliance from a local computer. On a NetScaler VPX appliance, you can only upload the license to the appliance from a local computer.
- You must purchase separate licenses for each appliance in a high availability (HA) pair. Make sure that the same type of licenses are installed on both the appliances. For example, if you purchase a platinum license for one appliance, you must purchase another platinum license for the other appliance.

This document includes the following information:

Prerequisites

Updated: 2014-06-24

To use the hardware serial number or license activation code to allocate your licenses:

- You must be able to access public domains through the appliance. The license allocation software internally accesses the Citrix licensing portal for your license. To access a public domain, you must configure a NetScaler IP (NSIP) address, configure a mapped IP (MIP) address or a subnet IP (SNIP) address, and set up a DNS server.
- Your license must be linked to your hardware, or you must have a valid license activation code (LAC). Citrix sends your LAC by email when you purchase a license.

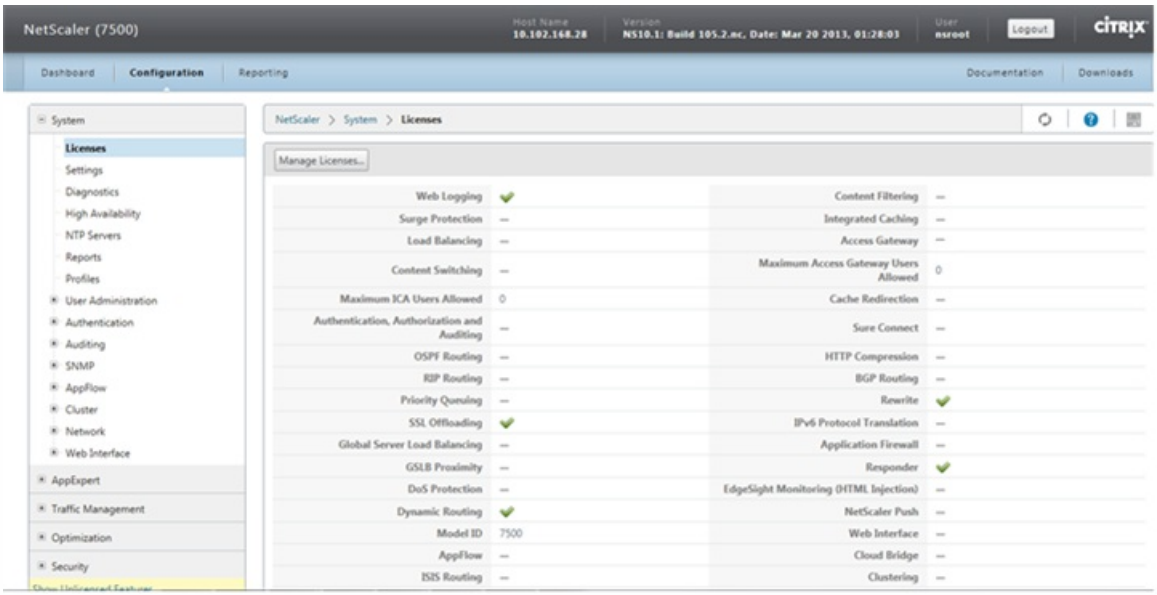
Allocating your License by using the Configuration Utility

Updated: 2014-02-11

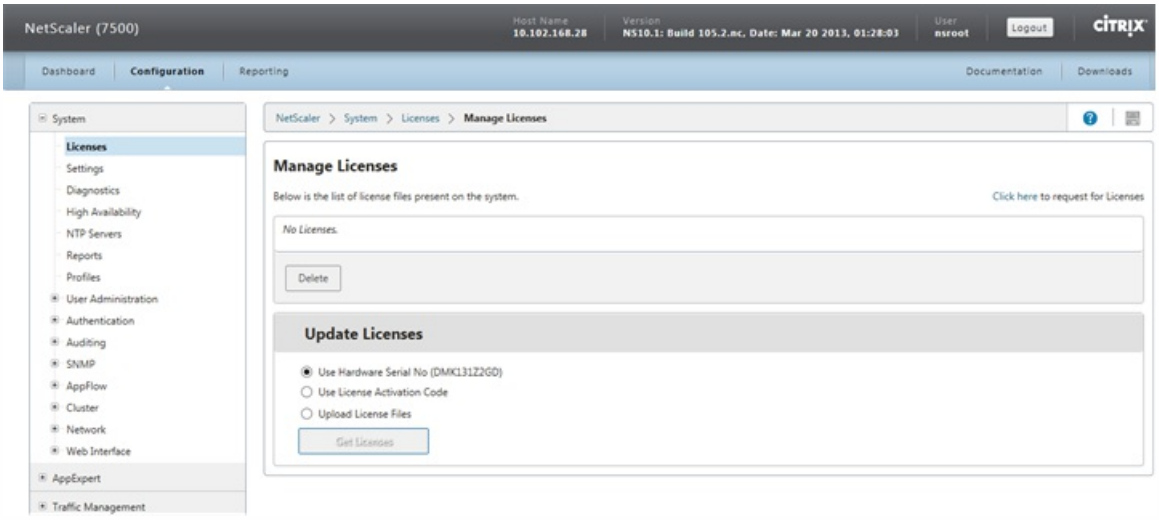
If your license is already linked to your hardware, the license allocation process can use the hardware serial number. Otherwise, you must type the license activation code (LAC).

To allocate your license

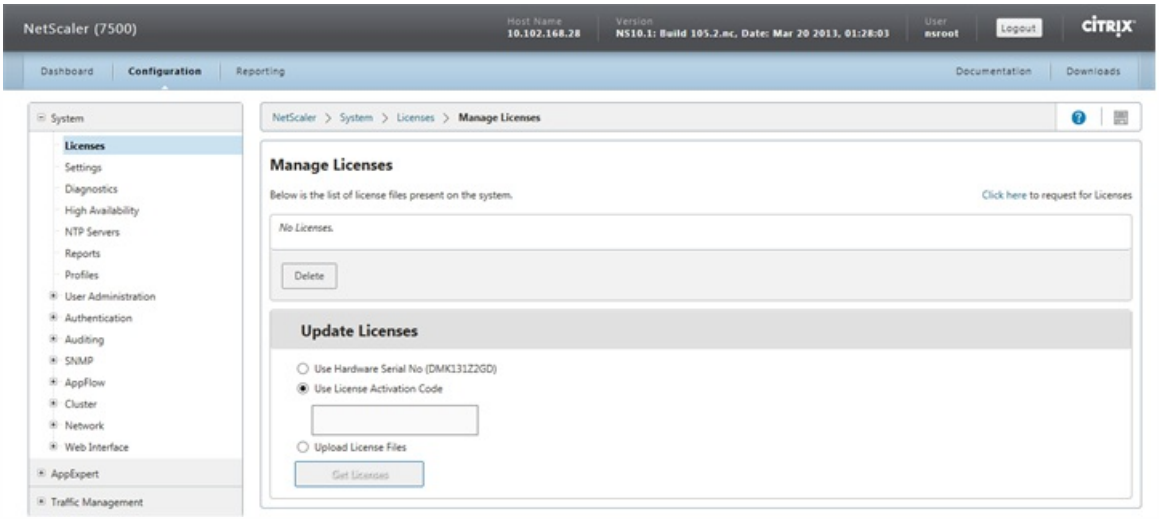
1. In a web browser, type the IP address of the NetScaler (for example, http://192.168.100.1).
2. In User Name and Password, type the administrator credentials.
3. On the Configuration tab, navigate to System > Licenses.
4. In the details pane, click Manage Licenses.



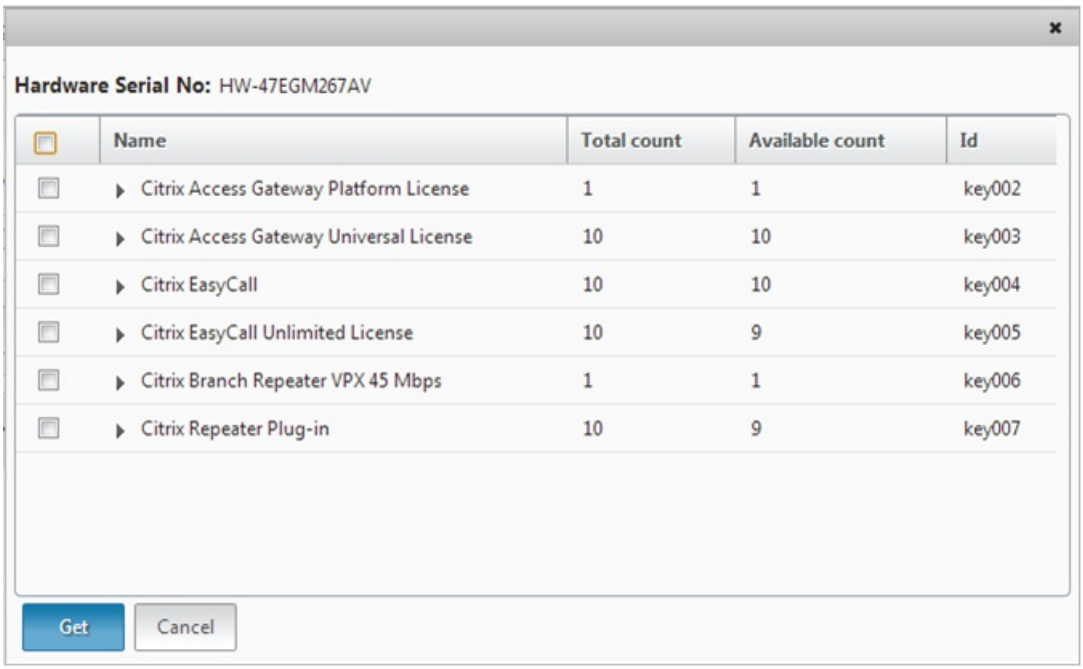
5. Click Update Licenses, and then select one of the following options:
   - **Use Hardware Serial Number**—The software internally fetches the serial number of your appliance and uses this number to display your license(s).



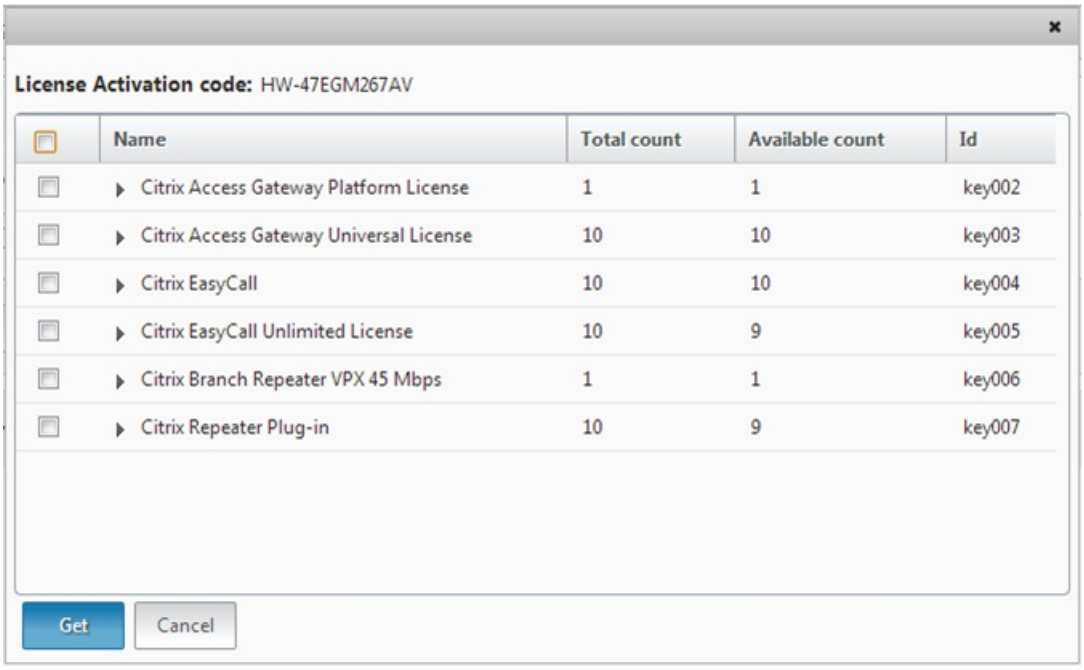
   - **Use License Activation Code**—Citrix emails the LAC for the license that you purchased. Enter the LAC in the text box.



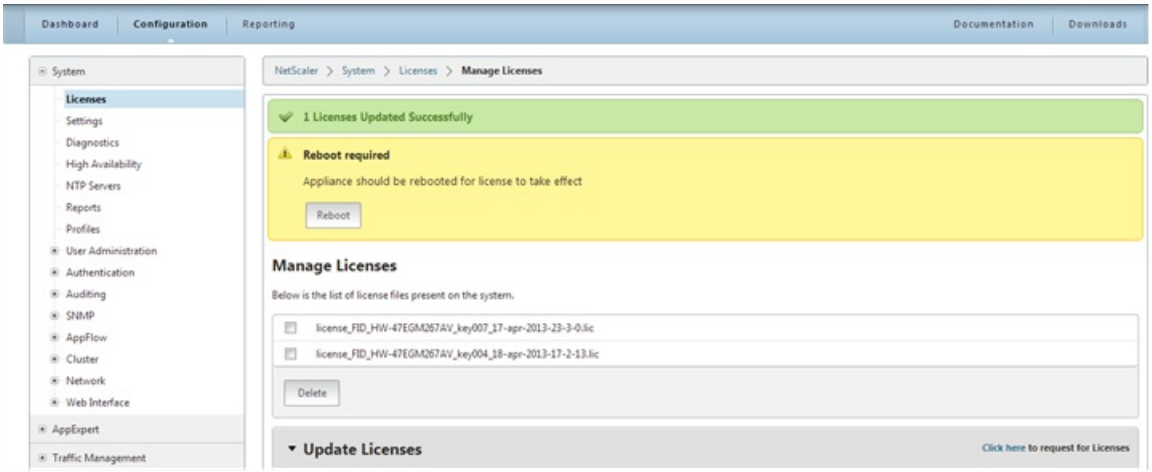
6. Click Get Licenses. Depending on the option that you selected, one of the following dialog boxes appears.
   - The following dialog box appears if you selected Hardware Serial Number.



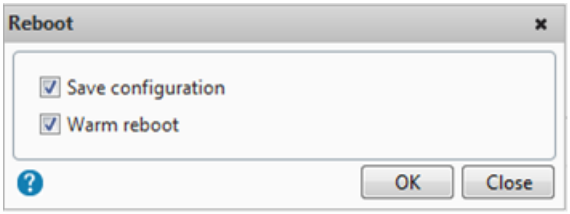
   - The following dialog box appears if you selected License Activation Code.



7. Select the license that you want to allocate, and then click Get.
8. Click Reboot for the license to take effect.



9. In the Reboot dialog box, click OK to proceed with the changes, or click Close to cancel the changes.



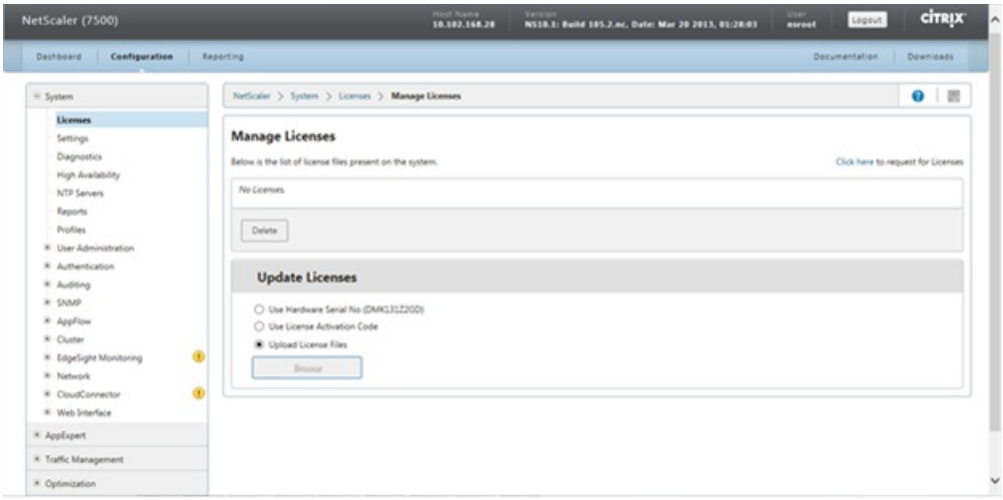
Installing the License

Updated: 2014-06-24

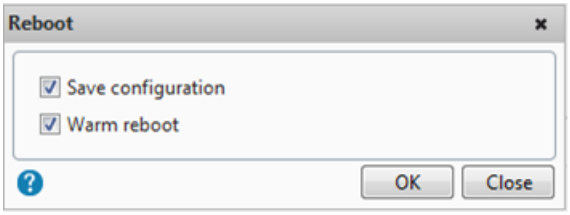
If you downloaded your license file to your local computer by accessing the licensing portal, you must upload the license to the appliance.

To install a license file by using the configuration utility

1. In a web browser, type the IP address of the NetScaler (for example, http://192.168.100.1).
2. In User Name and Password, type the administrator credentials.
3. On the Configuration tab, navigate to System > Licenses .
4. In the details pane, click Manage Licenses.
5. Click Update Licenses, and then select **Upload License Files**.



6. Click Browse. Navigate to the location of the license files, select the license file, and then click Open.
7. Click Reboot to apply the license.
8. In the Reboot dialog box, click OK to proceed with the changes, or click Close to cancel the changes.



See also

To install the licenses by using the command line interface

1. Open an SSH connection to the NetScaler by using an SSH client, such as PuTTY.
2. Log on to the NetScaler by using the administrator credentials.
3. Switch to the shell prompt, create a license subdirectory in the nsconfig directory, if it does not exist, and copy the new license file(s) to this directory.

Example

```
login: nsroot
Password: nsroot
Last login: Mon Aug  4 03:37:27 2008 from 10.102.29.9
Done
> shell
Last login: Mon Aug  4 03:51:42 from 10.103.25.64
root@ns# mkdir /nsconfig/license
root@ns# cd /nsconfig/license
```

Copy the new license file(s) to this directory.

Note: The NetScaler appliance does not prompt for a reboot option when you use the command line interface to install the licenses. Run the reboot -w command to warm reboot the system, or run the reboot command to reboot the system normally.

Verifying the Licensed Features

Updated: 2014-06-24

Before using a feature, make sure that your license supports the feature.

To verify the licensed features by using the command line interface

1. Open an SSH connection to the NetScaler by using an SSH client, such as PuTTY.
2. Log on to the NetScaler by using the administrator credentials.
3. At the command prompt, enter the sh ns license command to display the features supported by the license.

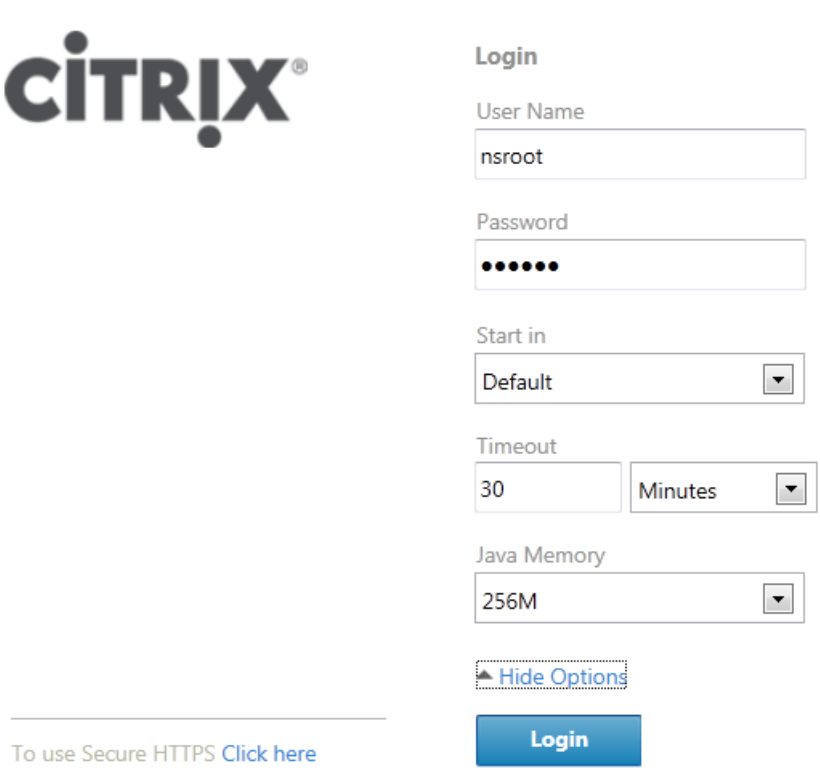
Example

```
sh ns license
      License status:
                  Web Logging: YES
              Surge Protection: YES
                .......

                HTML Injection: YES
 Done
```

To verify the licensed features by using the configuration utility

1. In a Web browser, type the IP address of the NetScaler, such as http://192.168.100.1.
2. In User Name and Password, type the administrator credentials.
3. In Start in, select Configuration, and then click Login, as shown in the following figure.

Figure 1. Login Screen



4. In the navigation pane, expand System, and then click Licenses. You will see a green check mark next to the licensed features.

Enabling or Disabling a Feature

Updated: 2014-07-01

When you use the NetScaler appliance for the first time, you need to enable a feature before you can use its functionality. If you configure a feature before it is enabled, a warning message appears. The configuration is saved but it will apply only after the feature is enabled.

To enable a feature by using the command line interface

At the NetScaler command prompt, type the following commands to enable a feature and verify the configuration:

- enable feature <FeatureName>
- show feature

Example

```
enable feature lb cs
done
 >show feature

      Feature                Acronym          Status
      -------                -------          ------
 1)   Web Logging             WL               OFF
 2)   Surge Protection         SP               ON
 3)   Load Balancing           LB               ON
 4)   Content Switching        CS               ON
 5)   Cache Redirection        CR               ON
```



```
.
.
.
24)  NetScaler Push          push          OFF
Done
```

The example shows how to enable load balancing (lb) and content switching (cs).

If the license key is not available for a particular feature, the following error message appears for that feature:

```
ERROR: feature(s) not licensed
```

Note: To enable an optional feature, you need a feature-specific license. For example, if you have purchased and installed the Citrix NetScaler Enterprise Edition license and need to enable the Integrated Caching feature, you first need to purchase and install the AppCache license.

To disable a feature by using the command line interface

At the NetScaler command prompt, type the following commands to disable a feature and verify the configuration:

- disable feature <FeatureName>
- show feature

Example

The following example shows how to disable load balancing (LB).

```
> disable feature lb
Done
> show feature

     Feature              Acronym        Status
     -------              -------        ------
1)   Web Logging          WL             OFF
2)   Surge Protection     SP             ON
3)   Load Balancing       LB             OFF
4)   Content Switching    CS             ON
.
.
.
24)  NetScaler Push       push           OFF
Done
>
```

Upgrading or Downgrading the System Software

Jun 25, 2014

NetScaler 10.1 offers new and updated features with increased functionality. A comprehensive list of enhancements is listed in the release notes accompanying the release announcement. Take a moment to read this document before you upgrade your software.

It is important to understand the licensing framework and types of licenses before you upgrade your software. A software edition upgrade may require new licenses, such as upgrading from the standard edition to the enterprise edition, the standard edition to the platinum edition, or the enterprise edition to the platinum edition.

Note: For upgrading or downgrading the nodes in a cluster setup, see "Upgrading or Downgrading the Cluster Software". Upgrading from release 10.1 build 121.10 or any earlier releases to release 10.1 build 122.17 and later involves some location changes of user monitor script files. For details, see Directory Locations of Script Files for User Monitors.

This document includes the following information:

- Upgrading to Release 10.1
- Upgrading to a Later Build within Release 10.1
- Downgrading from Release 10.1
- Downgrading to an Earlier Build within Release 10.1
- Auto Cleanup

Directory locations of script files for user monitors

Updated: 2014-06-24

In release 10.1 build 122.17, the script files for user monitors are at a new location. If you upgrade an appliance or virtual appliance to release 10.1 build 122.17 or later, the changes are as follows:

- A new directory named conflicts is created in /nsconfig/monitors/ and all the built-in scripts of the previous builds are moved to this directory.
- All new built-in scripts are available in the /netscaler/monitors/ directory. All custom scripts are available in the /nsconfig/monitors/ directory.
- You must save a new custom script in the /nsconfig/monitors/ directory.
- After the upgrade is completed, if a custom script is created and saved in the /nsconfig/monitors/ directory with the same name as that of a built-in script, the script in the /netscaler/monitors/ directory takes priority. That is, the custom script is not run.

If you provision a virtual appliance running release 10.1 build 122.17 or later, the changes are as follows:

- All built-in scripts are available in the /netscaler/monitors/ directory
- The directory /nsconfig/monitors/ is empty.
- If you create a new custom script, you must save it in the /nsconfig/monitors/ directory.

For more information about user monitors, see "Understanding User Monitors."

NetScaler Documentation

Apr 30, 2013

A complete set of NetScaler documentation is available on Citrix eDocs at http://support.citrix.com/proddocs/topic/netscaler/ns-gen-netscaler-wrapper-con.html.

You are encouraged to provide feedback and suggestions so that we can enhance the documentation.

Service and Support

Apr 30, 2013

Citrix® offers a variety of resources for support with your Citrix environment, including the following:

- The Knowledge Center is a self-service, Web-based technical support database that contains thousands of technical solutions, including access to the latest hotfixes, service packs, and security bulletins.
- Technical Support Programs for both software support and appliance maintenance are available at a variety of support levels.
- The Subscription Advantage program is a one-year membership that gives you an easy way to stay current with the latest product version upgrades and enhancements.
- Citrix Education provides official training and certification programs on virtually all Citrix products and technologies.

For more information about Citrix services and support, see the Citrix Systems Support Web site at http://www.citrix.com/lang/English/support.asp.

You can also participate in and follow technical discussions offered by the experts on various Citrix products at the following sites:

- http://community.citrix.com
- http://twitter.com/citrixsupport

FAQs

Feb 17, 2015

AppFlow

Jul 08, 2013

Which build of NetScaler supports AppFlow?

AppFlow is supported on NetScaler appliances running version 9.3 and above with nCore build.

What is the format used by AppFlow to transmit data?

AppFlow transmits information in the Internet Protocol Flow Information eXport (IPFIX) format, which is an open Internet Engineering Task Force (IETF) standard defined in RFC 5101. IPFIX (the standardized version of Cisco's NetFlow) is widely used to monitor network flow information.

What do AppFlow records contain?

AppFlow records contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. AppFlow records also contain application-level information (such as HTTP URLs, HTTP request methods and response-status codes, server response time, and latency). IPFIX flow records are based on templates that must be sent before sending flow records.

After an upgrade to NetScaler Version 9.3 Build 48.6 Cl, why does an attempt to open a virtual server from the GUI result in the error message "The AppFlow feature is only available on Citrix Netscaler Ncore"

AppFlow is supported only on nCore appliances. When you open the virtual server configuration tab, clear the **AppFlow** checkbox.

What does the transaction ID in an AppFlow records contain?

A transaction ID is an unsigned 32-bit number identifying an application-level transaction. For HTTP, a transaction corresponds to a request and response pair. All flow records that correspond to this request and response pair have the same transaction ID. A typical transaction has four uniflow records. If the NetScaler generates the response by itself (served from the integrated cache or by a security policy), there might be only two flow records for the transaction.

What is an AppFlow action ?

An Appflow action is a set of collectors to which the flow records are sent if the associated AppFlow policy matches.

What commands can I run on the NetScaler appliance to verify that the AppFlow action is a hit?

The show appflow action. For example:

```
> show appflow action
1)    Name: aFL-act-collector-1
      Collectors: collector-1
      Hits: 0
      Action Reference Count: 2
2)    Name: apfl-act-collector-2-and-3
      Collectors: collector-2, collecter-3
      Hits: 0
      Action Reference Count: 1
3)    Name: apfl-act-collector-1-and-3
      Collectors: collector-1, collecter-3
      Hits: 0
      Action Reference Count: 1
```

What is an AppFlow collector?

A collector receives flow records generated by the NetScaler appliance. To be able to send flow records, you must specify at least one collector. You can specify up to four. You can remove unused collectors.

What NetScaler verison is required for using AppFlow?

Use NetScaler version 9.3.49.5 or higher, and remember that AppFlow is available in only the nCore builds.

What transport protocol does AppFlow use?

AppFlow uses UDP as the transport protocol.

What ports needs to be opened if I have a firewall in the network?

Port 4739. It is the default UDP port the AppFlow collector uses for listening on IPFIX messages. If the user changes the default port, that port should be opened on the firewall.

How can I change the default port AppFlow uses?

When you add an AppFlow collector by using the add appflowCollector command, you can specify the port to be used.

```
> add appflowCollector coll1 -IPAddress
10.102.29.251 -port 8000
 Done
```

What does setting clientTrafficOnly do?

NetScaler generates AppFlow records only for client-side traffic.

How many collectors can be configured at a time?

You can configure up to four AppFlow collectors at a time on the NetScaler appliance. Please note that the maximum number of collectors that can be configured on a NetScaler appliance is four.

AutoScale

Sep 18, 2013

What are the prerequisites for setting up AutoScale?

For prerequisites for setting up AutoScale, see "Prerequisites".

Can the CloudPlatform AutoScale feature be used without a NetScaler appliance?

No. The NetScaler appliance is currently required for the AutoScale feature to work. If the CloudPlatform administrator configures AutoScale in a network that does not include a NetScaler appliance, CloudPlatform throws an error.

What happens if the AutoScale feature is used with a NetScaler release that does not support AutoScale?

If the AutoScale feature is used with a NetScaler release that does not support AutoScale, the CloudPlatform user interface throws an error. CloudPlatform also writes a message to the log file, indicating that the configured NetScaler does not support AutoScale.

What versions of CloudPlatform and NetScaler should I use to implement AutoScale?

For information about NetScaler releases that support AutoScale, see Supported Environment.

In a load balancing rule, can manually provisioned virtual machine instances coexist with instances provisioned by the AutoScale feature?

No. The CloudPlatform virtual machine group in a load balancing rule can contain only manually provisioned instances or only instances provisioned by the AutoScale feature. They cannot coexist.

Is there a limit on the number of virtual machine instances to which we can scale up by using AutoScale?

Yes. The CloudPlatform administrator specifies the maximum number of members to which the configuration can scale up. When the limit is reached, virtual machines are not provisioned even if the scale-up condition is satisfied. The upper limit prevents uncontrolled spawning of VMs due to misconfiguration of the AutoScale feature or unexpected load conditions.

Are AutoScale events observable?

The events generated for deploying or destroying virtual machines are observable. These events are logged in the NetScaler logs (ns.log) and in the CloudPlatform logs (management-server.log). However, you cannot observe the metric values collected by NetScaler monitors.

What metrics can be used in AutoScale policies?

In an AutoScale policy, you can use any metric that is exposed through SNMP, or any NetScaler statistics associated with the load balancing virtual server used in the AutoScale configuration. For example, you can use metrics associated with CPU, memory, or disk usage, and NetScaler metrics such as throughput or response time.

What should a CloudPlatform administrator do before performing maintenance tasks on a CloudPlatform network in which AutoScale is configured?

The CloudPlatform administrator should disable the AutoScale configuration from the CloudPlatform user interface. Disabling the AutoScale configuration temporarily disables any scale-up or scale-down events. However, disabling AutoScale for an application, in CloudPlatform, does not affect the ability of the NetScaler appliance to serve traffic to existing virtual machines.

With AutoScale configured, are any configured VM limits enforced on the user account?

The NetScaler appliance works in the context of an AutoScale user account. Therefore, any limits that the CloudPlatform

administrator has imposed on the number of VMs that can be created by the account are automatically enforced when the NetScaler appliance attempts to create more VMs than are permitted.

Is AutoScale supported in a high availability (HA) NetScaler pair?

No. Currently, HA mode is not supported for AutoScale.

Call Home

Jun 14, 2016

What is the Call Home feature on a NetScaler appliance?

The Call Home feature registers your NetScaler appliance with the Citrix Technical Support server (TaaS) and monitors the appliance for common error conditions. If your appliance is successfully registered with TaaS server, Call Home automatically uploads system debug data to that server in the event that one of the conditions occurs. The appliance keeps a full log of all upload events. If you are unable to correct the problem after reviewing the appliance's log, you can contact the Citrix Technical Support team and open a service request. The team can analyze the uploaded system data and recommend possible solutions.

Which release of NetScaler Software supports Call Home?

Release 10 and later.

Does Call Home support monitoring of any error conditions in a NetScaler virtual appliance?

No. Call Home does not currently support monitoring of virtual appliances.

Which NetScaler hardware models support Call Home?

Any NetScaler MPX appliance running release 10 or later of the NetScaler software supports Call Home.

Do you need a separate license for Call Home?

No. The Call Home feature does not require a separate license. It is available with all NetScaler platform licenses.

Does Call Home support monitoring of cluster events or error conditions?

No. Call Home does not currently support monitoring of NetScaler clusters.

What error conditions does Call Home monitor in a NetScaler appliance?

Call Home supports monitoring of the following events in a NetScaler appliance:

- Compact flash drive errors
- Hard disk drive errors
- Power supply unit failure
- SSL card failure
- Warm restart

What mechanism does Call Home use to upload the Call Home tar file to TaaS?

Call Home uses the HTTPS protocol to upload the Call Home tar file.

Does Call Home support automatic Technical Support service request creation?

No. You have to contact the Citrix Technical Support team to open a service request.

What is the frequency of Call Home tar file uploads to TaaS?

Call Home creates the Call Home tar file and uploads it to the Citrix Technical Support server (TaaS) upon first occurrence of a particular error condition since the appliance was last started. That is, a reoccurrence of same error condition does not trigger another upload unless the appliance was rebooted after the previous occurrence.

Must I configure SNMP for Call Home to monitor error conditions?

No. Call Home creates and uploads a Call Home tar file for the first occurrence of a monitored error condition since the

appliance was last started. If you want to be alerted each time the error condition occurs, you can configure the corresponding SNMP alarm for the error condition.

Is the Call Home feature enabled by default on the appliance?

No, the Call Home feature is disabled, by default. You must first enable the feature to register the appliance for critical error conditions.

Clustering

Feb 04, 2016

Click here for clustering FAQs for NetScaler versions 11.0, 10.5, and 10.1.

Configuration Utility - Frequently Asked Questions

Sep 05, 2014

Q: I am using a MAC Safari browser to upgrade a NetScaler ADC. On the upgrade wizard, when I click the Browse button to choose the build file from the appliance, the dialog box does not show any files or folders. Also, when I navigate back to the root folder, the dialog box displays the top level folder, but I cannot browse it. What should I do?

A: On the Safari browser, click the Settings icon and navigate to Preferences > Security > Manage Website Settings > Java, and then change value of the When visiting other websites setting to Run in unsafe mode.

Q: After I upgraded the JRE on my appliance to version 7.51, the graphical user interface (GUI) applet does not load when I access the NetScaler configuration utility. The applet download stops at 1%, and Java generates a security error. What should I do?

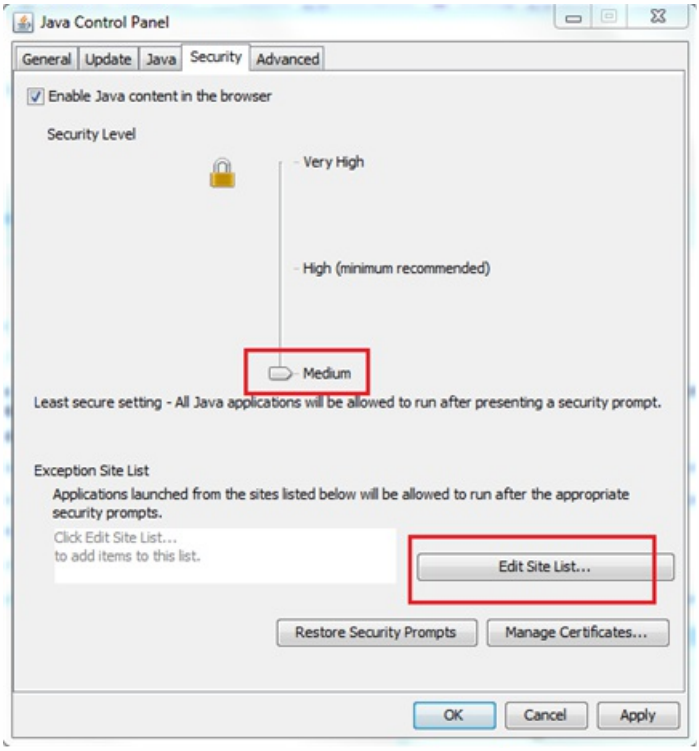
A: You can use the configuration utility with JRE 7.51 if, in the Java Control Panel, you lower the security level or add the NetScaler appliance's URL to the Exception Site List.

If using a Windows computer, navigate to Control Panel and click Java.

If using a MAC computer, navigate to System Preferences and click Java.

Then, in the Java Control Panel dialog box, do either of the following:

- Click the Security tab, and then set the Security Level to medium.
- Click the Security tab, click the Edit Site List button, and then add the URL of the NetScaler appliance in the Exception Site List box.
  Figure 1. Java Control Panel



Q: After I upgraded the JRE on my appliance to version 7.45, the graphical user interface (GUI) applet does not load when I access the NetScaler configuration utility. The applet download stops at 1%, and Java throws a security error. What should I do?

A: The NetScaler GUI is not compatible with JRE version 7.45. Citrix recommends using a JRE version earlier than 7.45 to access the configuration utility. If you have already upgraded to version 7.45 and do not want to downgrade, you can configure Java to not keep temporary files. However, you will have to download the JAR files every time you access the configuration utility.

To downgrade JRE 7.45 to an earlier version:

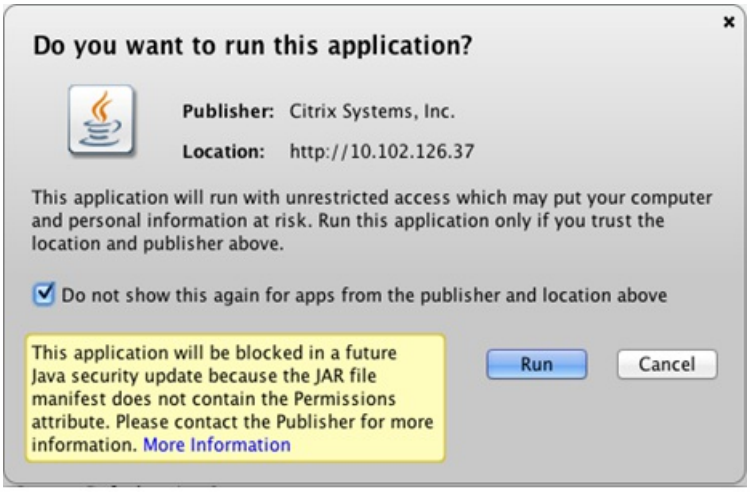
1. Uninstall JRE 7.45 by following the instructions at http://www.java.com/en/download/help/uninstall_java.xml.
2. Download an earlier version of JRE from the following location: http://www.oracle.com/technetwork/java/javase/archive-139210.html.

To continue using JRE 7.45:

If you are using a Windows system, navigate to Control Panel and click Java. The Java Control Panel opens.

If you are using a MAC system, navigate to System Preferences and click Java. The Java Control Panel opens.

1. Click the General tab.
2. Under Temporary Internet Files, click Settings.
3. Clear the Keep temporary files on my computer option.
4. Close the browser and re-launch the GUI.
5. Open the configuration utility and click Run and Allow, respectively, when the following warnings appear:



Note:

- The Jar files will not be cached and you will have to download the files every time you access the configuration utility.
- This problem is fixed in the following releases:
  - Release 9.3, build 65.x and later
  - Release 10.0, build 78.x and later
  - Release 10.1, build 122.x and later
  - Release 10.1.e, build 120.13xx.e

Q: What should I do before accessing the NetScaler configuration utility?

A: Before accessing a new version of the NetScaler software:

- Clear your browser cache.
- Make Sure that JavaScript, Java, and plug-ins are enabled in your browser. For help with enabling Java for your browser, see http://java.com/en/download/help/enable_browser.xml.
- Clear the "Temporary internet files" in the Java console.
- On the Java tab of the Java console, in Java Runtime Environment Settings, make sure that the latest version of JRE is present and is enabled.

Q: I am using HTTP to access the configuration utility. Which port should I open?

A: Open TCP port 3010 when using HTTP to access the configuration utility.

Q: I am using HTTPS to access the configuration utility. Which port should I open?

A: Open TCP port 3008 when using HTTPS to access the configuration utility.

Q: After entering the IP address of the NetScaler appliance in the address bar, I get the following error: "Java Applet could not be loaded". What should I do?

A: Verify that Java is installed properly. You can download Java from www.java.com. If you are using a MAC Safari browser, Java is disabled if it is not used for 35 days.

To enable Java plug-in in Safari, follow these steps:

1. In the Safari browser, choose Safari > Preferences or press Command-comma (⌘-,) on your keyboard.
2. Click Security, and then select Enable Java.

3. Close the Safari Preferences window.

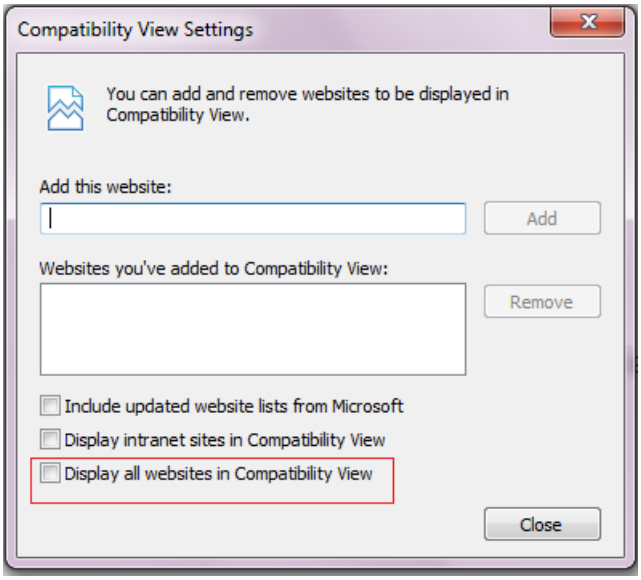
Q: With which browsers is the configuration utility compatible for different operating systems?

A: The following table lists the compatible browsers:

Operating System	Browser	Versions
Windows 7	Internet Explorer	8 and 9
	Mozilla Firefox	3.6.25 and above
	Chrome	15 and above
Windows 64 bit	Internet Explorer	8 and 9
	Chrome	15 and above
MAC OS	Mozilla Firefox	12 and above
	Safari	5.1.3

Q: When I access the NetScaler configuration utility by using Internet Explorer version 8 or 9, the browser displays only a grey bar at the top of the screen. What should I do?

A: The browser might be set in compatibility mode. To disable compatibility mode, go to **Tools** > **Compatibility View Settings** and clear the **Display all websites in Compatibility View** check box.



Q: Even after I disable compatibility mode in Internet Explorer version 8 or 9, the configuration utility does not appear. What should I do?

A: Make sure that the browser mode and document mode in the browser are set to the same version. To view the configuration, press F12. Set the values to either Internet Explorer 8 or Internet Explorer 9.

Q: When I access the NetScaler configuration utility by using Internet Explorer version 9, the utility displays the following error message: "You are not logged in. Please login." What should I do?

A: Make sure that the cookies are not blocked in your Interner Explorer settings. Go to **Tools** > **Internet Options**. Click the **Privacy** tab, and then under **Settings**, make sure that the slider is set to **Medium** or any lower value.

Q: I am using a MAC OS with JRE 1.7. After logging on to the configuration utility, I am not able to enter value in any of the text fields. What should I do?

A: Install Java 7, update 21 or higher.

Q: I am using a MAC OS. When I click outside a dialog window, the screen goes out of focus. Now, my browser looks disabled and hung. What should I do?

A: Click on the Java icon in the system dock, and in the JRE Security Warning window, click **Don't Block**. For details, see http://www.oracle.com/technetwork/java/javase/7u21-relnotes-1932873.html

Q: Is there any compatibility issue in using JRE version 7_11 with the latest version of browsers?

A: Yes. Internet Explorer 9 and Firefox 18.0.1 block Java on computers running JRE version 7_11. You have to manually activate Java in the browser or upgrade to a later version of JRE (JRE 7_13).

Content Switching

Aug 30, 2013

I have installed a non-NetScaler load balancing appliance on the network. However, I would like to use the content switching feature of the NetScaler appliance to direct the client requests to the load balancing appliance. Is it possible to use the Content switching feature of the NetScaler appliance with a non-NetScaler load balancing appliance?

Yes. You can use the Content switching feature of the NetScaler appliance with the load balancing feature of the NetScaler appliance or a non-NetScaler load balancing appliance. However, when using the non-NetScaler load balancing appliance, make sure that you create a load balancing virtual server on the NetScaler appliance and bind it to the non-NetScaler load balancing appliance as a service.

How is a Content switching virtual server different from a load balancing virtual server?

A Content switching virtual server is capable only of sending the client requests to other virtual servers. It does not communicate with the servers.

A Load balancing virtual server balances the client load among servers and communicates with the servers. It monitors server availability and can be used to apply different load balancing algorithms to distribute the traffic load.

Content switching is a method used to direct client requests for specific types of content to targeted servers by way of load balancing virtual servers. You can direct the client requests to the servers best suited to handle them. This result in reduced overheads to process the client requests on the servers.

I want to implement the Content switching feature of the NetScaler appliance to direct the client requests. What types of client request can I direct by using the Content switching feature?

You can direct only HTTP, HTTPS, FTP, TCP, Secure TCP, and RTSP client requests by using the Content switching feature. To direct HTTPS client requests, you must configure the SSL offload feature on the appliance.

I want to create Content switching rules on the NetScaler appliance. What are the various elements of the client request on which I can create a content switching rule?

You can create the content switching rules based on the following elements and their values in the client request:

- URL
- URL tokens
- HTTP version
- HTTP Headers
- Source IP address of the client
- Client version
- Destination TCP port

I understand that the content switching feature of the NetScaler appliance helps enhance the performance of the network. Is this correct?

Yes. You can direct the client requests you the servers best suited to handle them. The result is reduced overhead for processing the client requests on the servers.

Which feature of the NetScaler appliance should I configure on the NetScaler appliance to enhance the site manageability and response time to the client requests?

You can configure the content switching feature of the NetScaler appliance to enhance the site manageability and

response time to the client request. This feature enables you to create content groups within the same domain name and IP address. This approach is flexible, unlike the common approach of explicitly partitioning the content into different domain names and IP addresses, which are visible to the user.

Multiple partitions dividing a Web site into various domain names and IP addresses force the browser to create a separate connection for each domain it finds when rendering and fetching the content of a web page. These additional WAN connections degrade the response time for the web page.

I have hosted a web site on a web server farm. What advantages does the NetScaler content switching feature offer for this type of setup?

The content switching feature provides the following advantages on a NetScaler appliance in in a site that is based in a web server farm:

- Manage the site content by creating a content group within the same domain and IP address.
- Enhance the response time to client requests by using the content group within the same domain and IP address.
- Avoid the need for full content replication across domains.
- Enable application-specific content partitioning. For example, you can direct client requests to a server that handles only dynamic content or only static content, as appropriate for the request.
- Support multi-homing of multiple domains on the same server and use the same IP address.
- Reuse connections to the servers.

I want to implement the content switching feature on the NetScaler appliance. I want to direct the client requests to the various servers after evaluating the various parameters of each request. What approach should I follow to implement this setup when configuring the content switching feature?

You can use policy expressions to create policies for the content switching feature. An expression is a condition evaluated by comparing the qualifiers of the client request to an operand by using an operator. You can use the following parameters of the client request to create an expression:

- **Method**- HTTP request method.
- **URL**- URL in the HTTP header.
- **URL TOKENS**- Special tokens in the URL.
- **VERSION**- HTTP request version.
- **URL QUERY**- Contains the URL Query LEN, URL LEN, and HTTP header.
- **SOURCEIP**- IP address of the client.

Following is a complete list of the operators that you can use to create an expression:

- == (equals)
- != (not equals)
- EXISTS
- NOT EXISTS
- CONTAINS
- NOT CONTAINS
- GT (greater than)
- LT (less than)

You can also create various rules, which are logical aggregations of a set of expressions. You can combine multiple expressions to create rules. To combine expressions, you can use the && (AND) and || (OR) operators. You can also use parenthesis to create nested and complex rules.

I want to configure a rule based policy along with a URL based policy for the same content switching virtual

server. Is it possible to create both types of policies for the same content switching virtual server?

Yes. You can create both type of policies for the same content switching virtual server. However, be sure to assign priorities to set an appropriate precedence for the policies.

I want to create content switching policies that evaluate the domain name, along with a prefix and suffix of a URL, and direct the client requests accordingly. Which type of content switching policy should I create?

You can create a Domain and Exact URL policy. When this type of policy is evaluated, the NetScaler appliance selects a content group if the complete domain name and the URL in the client request match the ones configured. The client request must match the configured domain name and exactly match the prefix and suffix of the URL if they are configured.

I want to create content switching policies that evaluate the domain name, along with a partial prefix and suffix of URL, and direct the client requests accordingly. Which type of content switching policy should I create?

You can create a Domain and Wildcard URL policy for the content switching virtual server. When this type of policy is evaluated, the NetScaler appliance selects a content group if the request matches the complete domain name and partially matches the URL prefix.

What is a Wildcard URL policy?

You can use wildcards to evaluate partial URLs in client requests to the URL you have configured on the NetScaler appliance. You can use wildcards in the following types of URL-based policies:

- Prefix only. For example, the /sports/* expression matches all URLs available under the /sports URL. Similarly, the /sports* expression matches all URLs whose prefix is /sports .
- Suffix only. For example the /*.jsp expression matches all URLs with a file extension of jsp.
- Prefix and Suffix. For example, the /sports/*.jsp expression matches all URLs under the /sports/ URL that also have the jsp file extension. Similarly, the /sports*.jsp expression matches all URLs with a prefix of /sports* and a file extension of jsp.

What is a Domain and Rule policy?

When you create a Domain and Rule policy, the client request must match the complete domain and the rule configured on the NetScaler appliance.

What is the default precedence set for evaluating policies?

By default, the rule based policies are evaluated first.

If some of the content is the same for all client requests, what type of precedence should I use for evaluating policies?

If some of the content is same for all the users and different content should be served on the basis of client attributes, you can use URL-based precedence for policy evaluation.

What policy expression syntaxes are supported in content switching?

Content switching supports two types of policy expressions:

- **Classic Syntax-** Classic syntax in content switching starts with the keyword REQ and is more advanced than the default syntax. Classic policies cannot be bound to an action. Therefore, the target load balancing virtual server can be added only after binding the content switching virtual server.
- **Default Syntax:** Default syntax generally starts with key word HTTP and is easier to configure. A target load balancing virtual server action can be bound to a Default Syntax policy, and the policy can be used on multiple content switching virtual servers.

Can I bind a single content switching policy to multiple virtual servers?

Yes. You can bind a single content switching policy to multiple virtual servers by using policies with defined actions. Content switching policies that use an action can be bound to multiple content switching virtual servers because the target load balancing virtual server is no longer specified in the content switching policy. The ability to bind a single policy to multiple content switching virtual servers helps to further reduce the size of the content switching configuration.

For more information, see the following Knowledge Center articles and eDocs topics:

- "How to Bind the Same Content Switching Policy to Two Content Switching vServers on a NetScaler Appliance," at http://support.citrix.com/article/CTX122918 .
- "How to Bind the Same Advanced Policy to Multiple Content Switching Virtual Servers using Policy Labels," at http://support.citrix.com/article/CTX122736.
- "Configuring a Content Switching Action," at http://support.citrix.com/proddocs/topic/netscaler-traffic-management-10-map/ns-cs-basicconfig-config-cs-action-con.html

Can I create an action based policy using classic expressions?

No. As of now NetScaler does not support policies using classic syntax expressions with actions. The target load balancing virtual server should be added when binding the policy instead of defining it in an action.

High Availability

Sep 30, 2013

What are the various ports used to exchange the HA-related information between the nodes in an HA configuration?

In an HA configuration, both nodes use the following ports to exchange HArelated information:

- UDP Port 3003, to exchange heartbeat packets
- Port 3010, for synchronization and command propagation

What configurations are not synced or propagated in an HA configuration in either INC or non-INC mode?

Configurations implemented with the following commands are neither propagated nor synced to the secondary node:

- All node specific HA configuration commands. For example, add ha node, set ha node, and bind ha node.
- All Interface related configuration commands. For example, set interface and unset interface.
- All channel related configuration commands. For example, add channel, set channel, and bind channel.

Note: For more information about HA Configuration in INC mode, see Configuring High Availability Nodes in Different Subnets.

What configurations are not synced or propagated in an HA configuration in INC mode?

The following configurations are neither synced nor propagated. Each node has its own.

- MIPs
- SNIPs
- VLANs
- Routes (except LLB routes)
- Route monitors
- RNAT rules (except any RNAT rule with VIP as the NAT IP)
- Dynamic routing configurations.

What are the conditions that trigger synchronization?

Synchronization is triggered by any of the following conditions:

- The incarnation number of the primary node, received by the secondary, does not match that of the secondary node. Note: Both nodes in an HA configuration maintain a counter called *incarnation number*, which counts the number of configurations in the node's configuration file. Each node sends its incarnation number to each other node in the heartbeat messages. The incarnation number is not incremented for the following commands:
  - All HA configuration related commands. For example, add ha node, set ha node, and bind ha node.
  - All Interface related commands. For example, set interface and unset interface.
  - All channel-related commands. For example, add channel, set channel, and bind channel.
- The secondary node comes up after a restart.
- The primary node becomes secondary after a failover.

Does a configuration added to the secondary node get synchronized on the primary?

No, a configuration added to the secondary node is not synchronized to the primary.

What could be the reason for both nodes claiming to be the primary in an HA configuration?

The most likely reason is that the primary and secondary nodes are both healthy but the secondary does not receive the heartbeat packets from the primary. The problem could be with the network between the nodes.

Does an HA configuration run into any issues if you deploy the two nodes with different system clock settings?

Different system-clock settings on the two nodes can cause the following issues:

- The time stamps in the log file entries do not match. This situation makes it difficult to analyze the log entries for any issues.
- After a failover, you might have problems with any type of cookie based persistence for load balancing. A significant difference between the times can cause a cookie to expire sooner than expected, resulting in termination of the persistence session.
- Similar considerations apply to any time related decisions on the nodes.

What are the conditions for failure of the *force HA sync* command?

Forced synchronization fails in any of the following circumstances:

- You force synchronization when synchronization is already in progress.
- The secondary node is disabled.
- HA synchronization is disabled on the current secondary node.
- HA propagation is disabled on the current primary node and you force synchronization from the primary.

What are the conditions for failure of the *sync HA files* command?

Synchronizing configuration files fail if the secondary node is disabled.

In an HA configuration, if the secondary node takes over as the primary, does it switch back to secondary status if the original primary comes back online?

No. After the secondary node takes over as the primary, it remains as primary even if the original primary node comes back online again. To interchange the primary and secondary status of the nodes, run the *force failover* command.

What are the conditions for failure of the *force failover* command?

A forced failover fails in any of the following circumstances:

- The secondary node is disabled.
- The secondary node is configured to remain secondary.
- The primary node is configured to remain primary.
- The state of the peer node is unknown.

Hardware FAQs

Dec 04, 2014

Transceivers

Are transceivers shipped with the MPX 8005/8015/8200/8400/8600/8800 appliance?

No. Transceivers are available for purchase separately. Contact your Citrix sales representative to order transceivers for your appliance.

Are transceivers hot-swappable?

The 1G SFP transceiver is hot-swappable with release 9.3 build 47.5 or later on the following NetScaler appliances, which use the Intel e1k interface:

- MPX 7500/9500
- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500
- MPX 11500/13500/14500/16500/18500/20500
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150

The 10G SFP+ transceiver is hot-swappable with release 9.3 build 57.5 or later on the following NetScaler appliances, which use the ixgbe (ix) interface:

- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500
- MPX 11500/13500/14500/16500/18500/20500
- MPX 17500/19500/21500
- MPX 17550/19550/20550/21550
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150

Why does the 10G SFP+ transceiver autonegotiate to 1G speed?

Autonegotiation is enabled by default on the 10G SFP+ ports into which you insert your 10G SFP+ transceiver. When a link is established between the port and the network, the speed is autonegotiated. For example, if you connect the port to a 1G network, the speed is autonegotiated to 1G.

Can I insert a 1G transceiver into a 10G slot?

The 10G slot supports copper 1G transceivers, which can operate at up to 1 Gbps in a 10 Gbps slot.

Note that you cannot insert a 10G transceiver into a 1G slot.

The following table shows the compatibility matrix of transceivers and ports available on the NetScaler appliance.

Ports	Transceivers		
	10G	1G Fiber	1G Copper

10G	Supported	Not Supported	Supported
1G Fiber	Not Supported	Supported	Not Supported
1G Copper	Not Supported	Not Supported	Supported

What is QSFP+?

QSFP+ stands for Quad Small Form-factor Pluggable, which is a small, hot-pluggable transceiver for connecting data devices. This transceiver is used for 40G interfaces.

QSFP+ to Four SFP+ Copper Breakout Cables—These cables connect to four SFP+ 10GE ports of a NetScaler appliance on one end and to a QSFP+ 40G port of a Cisco switch on the other end.

Support for 40G connectivity—NetScaler models that have at least four 10G SFP+ ports connect to Cisco 40G interfaces by aggregating four of the 10G SFP+ ports to form a 40G link aggregation channel. QSFP to Four port SFP+ Copper Breakout Cable **QSFP-4SFP10G-CU3M (reports as L45593-D178-C30)** is used.

Which NetScaler appliances support the QSFP-4SFP10G-CU3M (reports as L45593-D178-C30) Breakout Cable?

NetScaler appliances that have at least four 10G SFP+ ports support this cable. The following appliances have at least four 10G SFP+ ports:

- MPX 11500/13500/14500/16500/18500/20500
- MPX 17550/19550/20550/21550
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150

QSFP-4SFP10G-CU3M breakout cable is supported by NetScaler release 9.3 build 65.8 or later, and release 10.1 build 122.17 or later.

Power Supplies

Is the power supply on the NetScaler MPX 5500 and MPX 5550/5650 appliances field replaceable?

No. The power supply on the NetScaler MPX 5500 and MPX 5550/5650 appliances is fixed.

Does the MPX 8005/8015/8200/8400/8600/8800 appliance ship with two power supplies?

No. The MPX 8005/8015/8200/8400/8600/8800 appliance supports dual power supplies but ships with one power supply. Contact your Citrix sales representative to order a second power supply.

How many power supplies are shipped with each platform?

The following table lists the number of power supplies shipped with each platform:

Platform	Number of Power Supplies shipped
MPX 5500	1
MPX 7500/9500	1 (You can order a second power supply.)

Platform	Number of Power Supplies shipped
MPX 9700/10500/12500/15500	2
MPX 15000/17000	1 (You can order a second power supply.)
MPX 11500/13500/14500/16500/18500/20500	2
MPX 17500/19500/21500	1 (You can order a second power supply.)
MPX 17550/19550/20550/21550	2

Are power supplies hot-swappable?

Yes. If the appliance has two power supplies, you can replace one power supply without shutting down the appliance, provided the other power supply is working.

Rack and Rail

Do you have different rail kits for 1U and 2U appliances?

No. All MPX and SDX appliances use the same rail kit. The kit contains two pairs of slide rails, of different lengths, for a 1U and a 2U appliance.

Which rail kit should I buy?

The appliance ships with the standard 4-post rail kit that fits racks from 28-38 inches.

The compact 4-post rail kit for racks from 23-33 inches, or the 2-post rail kit for 2-post racks, has to be purchased separately. Contact your Citrix sales representative to order the appropriate kit.

What are the maximum and the minimum lengths of the outer rack rails?

The length of a standard outer rack rail is from 28 to 38 inches. The length of a shorter outer rack rail is from 23 to 33 inches.

What is the space required between the front post and rear post of the rack?

Standard racks require 28–38 inches between the front and rear posts. Shorter racks require from 23 to 33 inches.

How far can an appliance extend from the front post of the rack?

The chassis can extend up to 1.25 inches from the front post for all NetScaler MPX and SDX appliances.

How much space is required for maintaining the front and rear area of an appliance?

Minimum clearance areas of 36 inches for the front area and 24 inches for the rear area are required for maintenance of all NetScaler MPX and SDX appliances.

Lights Out Management (LOM) Port

Which LOM features are supported on the NetScaler MPX Appliance?

The MPX 8005/8015/8200/8400/8600/8800, MPX 11500/13500/14500/16500/18500/20500, and MPX 17550/19550/20550/21550 have an Intelligent Platform Management Interface (IPMI), also known as the Lights out Management (LOM) port, on the front panel of the appliance. The following three LOM features are supported on those

platforms:

- Configuring the LOM port
- Power cycling the appliance
- Performing a core dump

Can the LOM interface be configured to accept only encrypted Virtual Network Computer (VNC) sessions on TCP port 5900?

Yes, customers who enable Transport Layer Security (TLS) on their LOM interface will have their VNC connections delivered over TLS as well.

For more information on LOM security guidelines, see Secure Deployment Guide for NetScaler MPX, VPX, and SDX Appliances.

Can the version of SSH used on the LOM interface be upgraded? Is there a patch available?

Individual components of the LOM cannot be upgraded independently. You must upgrade the entire LOM firmware as a package. The latest available LOM package can be found on the Citrix downloads website under LOM Firmware Upgrade.

Is it possible to add a third-party or self-signed SSL certificate to the LOM interface?

Yes, you can enable SSL on the latest binaries for third-party and self-signed SSL certificates, except on the 88XX models. On those models, the current LOM release does not support third-party certificates.

General

What is the recommended terminal emulator?

PuTTY.

Which platforms support Pay-As-You-Grow licenses?

The following platforms support Pay-As-You-Grow licenses:

- MPX 5550 to MPX 5650
- MPX 7500 to MPX 9500
- MPX 8005 to MPX 8015 to MPX 8200 to MPX 8400 to MPX 8600 to MPX 8800
- MPX 11500 to MPX 13500 to MPX 14500 to MPX 16500 to MPX 18500 to MPX 20500
- MPX 17500 to MPX 19500 to MPX 21500
- MPX 17550 to MPX 19550 to MPX 20550 to MPX 21550
- MPX 22040 to MPX 22060 to MPX 22080 to MPX 22100 to MPX 22120

Do you support direct attach cable (DAC)?

Yes, Citrix NetScaler appliances support a passive DAC in the following releases and builds:

- Release 9.3, build 63.4 and later
- Release 9.3.e, build 60.3007.e and later
- Release 10, build 74.2 and later
- Release 10.1, build 112.15 and later

Which port should I insert the DAC into?

DAC is inserted into the 10G port on the appliance.

Does the 1G port support DAC?

No. The DAC might fit into a 1G port but is not supported.

How can I order a DAC?

Contact your Citrix sales representative to order a DAC.

Can I mix DAC and fiber transceivers on the same appliance?

Yes. You can mix DAC and fiber transceivers on the same appliance. Each 10G port supports both options.

Can I mix SFP+ fiber and DAC in ports that are part of the same link aggregation channel (LAC)?

No. There must be symmetry between all elements in the same LAC.

Integrated Caching

Feb 28, 2014

Content Groups

How is a DEFAULT content group different from other content groups?

The behavior of the DEFAULT content group is exactly the same as any other group. The only attribute that makes the DEFAULT content group special is that if an object is being cached and no content group has been created, the object is cached in the DEFAULT group.

What is the 'cache-Control' option of the content group level?

You can send any cache-control header the browser. There is a content group level option, -cacheControl, which enables you to specify the cache-control header that you want to be inserted in the response to the browser.

What is the 'Minhit' option in content group level?

Minhit is an integer value specifying the minimum number of hits to a cache policy before the object is cached. This value is configurable at the content group level. Following is the syntax to configure this value from the command line interface.
add/set cache contentGroup <Content_Group_Name> [-minHits <Integer>]

What is the use of the expireAtLastByte option?

The expireAtLastByte option enables the integrated cache to expire the object as soon as it has been downloaded. Only requests that are outstanding requests at that time are served from cache. any new requests are sent to the server. This setting is useful when the object is frequently modified, as in the case of stock quotes. This expiry mechanism works along with the Flash Cache feature. To configure expireAtLastByte option, run the following command from the command line interface:

add cache contentGroup <Group_Name> –expireAtLastByte YES

Cache policy

What is a caching policy?

Policies determine which transactions are cacheable and which are not. Additionally, policies add or override the standard HTTP caching behavior. Policies determine an action, such as CACHE or NOCACHE, depending on the specific characteristics of the request or response. If a response matches policy rules, the object in the response is added to the content group configured in the policy. If you have not configured a content group, the object is added to the DEFAULT content group.

What is a policy hit?

A hit occurs when a request or response matches a cache policy.

What is a miss?

A miss occurs when a request or response does not match any cache policy. A miss can also occur if the request or response matches a cache policy but some override of RFC behavior prevents the object from being stored in the cache.

I have configured Integrated Caching feature of the NetScaler appliance. When adding the following policy, an error message appears. Is there any error in the command?

```
add cache policy image_caching -rule exp1 | ns_ext_not_jpeg –action cache
> ERROR: No such command
```

In the preceding command, the expression should be within the quotation marks. Without quotation marks, the operator is considered to be the pipe operator.

Memory Requirements

What are the commands that I can run on the NetScaler appliance to check the memory allocated to cache?

To display the memory allocated for cache in the NetScaler appliance, run any of the following commands from the command line interface:

- show cache parameter
  In the output, check the value of the Memory usage limit parameter. This is the maximum memory allocated for cache.
- show cache <Content_Group_Name>
  In the output, check the values of the Memory usage and Memory usage limit parameters indicating the memory used and allocated for the individual content group.

My NetScaler appliance has 2 GB of memory. Is there any recommended memory limit for cache?

For any model of the NetScaler appliance, you can allocate half of the memory to the cache. However, Citrix recommends allocating a little less than half of the memory, because of internal memory dependency. You can run the following command to allocate 1 GB of memory to cache:

```
set cache parameter -memLimit 1024
```

Is it possible to allocate memory for individual content groups?

Yes. Even though you allocate memory for the integrated cache globally by running the set cache parameter –memlimit<Integer>, you can allocate memory to individual content groups by running the set cache <Content_Group_Name> –memLimit <Integer> command. The maximum memory you can allocate to content groups (combined) cannot exceed the memory you have allocated to the integrated cache.

What is the dependency of memory between integrated cache and TCP buffer?

If the NetScaler appliance has 2 GB memory, then the appliance reserves approximately 800 to 900 MB of memory and remaining is allocated to FreeBSD operating system. Therefore, you can allocate up to 512 MB of memory to integrated cache and the rest is allocated to TCP buffer.

Does it affect the caching process if I do not allocate global memory to the integrated cache?

If you do not allocate memory to integrated cache, all requests are sent to the server. To make sure that you have allocated memory to the integrated cache, run the show cache parameter command. Actually no objects will be cached if global memory is 0, so this needs to be set first.

Verification commands

What are the options for displaying cache statistics?

You can use either of the following options to display the statistics for cache:

- stat cache
  To display the summary of the cache statistics.

- stat cache –detail
  To display the full details of the cache statistics.

What are the options for displaying the cached content?

To display the cached content, you can run the show cache object command.

What is the command that I can run to display the characteristics of an object stored in cache?

If the object stored in the cache is, for example, GET //10.102.12.16:80/index.html, you can display the details about the object by running the following command from the command line interface of the appliance:

```
show cache object -url '/index.html' -host 10.102.3.96 -port 80
```

Is it mandatory to specify the group name as a parameter to display the parameterized objects in cache?

Yes. It is mandatory to specify the group name as a parameter to display the parameterized objects in cache. For example, consider that you have added the following policies with the same rule:
add cache policy p2 -rule ns_url_path_cgibin -action CACHE –storeInGroup g1
add cache policy p1 -rule ns_url_path_cgibin -action CACHE -storeInGroup g2
In this case, for the multiple requests, if policy p1 is evaluated, its hit counter is incremented and the policy stores the object in the g1 group, which has hit parameters. Therefore, you have to run the following command to display the objects from the cache:

show cache object -url "/cgi-bin/setCookie.pl" -host 10.102.18.152 groupName g1

Similarly, for another set of multiple requests, if policy p2 is evaluated, its hit counter is incremented and the policy stores the object in the g2 group, which does not have hit parameters. Therefore, you have to run the following command to display the objects from the cache:

show cache object -url "/cgi-bin/setCookie2.pl" -host 10.102.18.152

I notice that there are some blank entries in the output of the nscachemgr command. What are those entries?

Consider the following sample output of the nscachemgr command. The blank entries in this output are highlighted in bold face for your reference:

```
root@ns# /netscaler/nscachemgr -a
//10.102.3.89:80/image8.gif
//10.102.3.97:80/staticdynamic.html
//10.102.3.97:80/
//10.102.3.89:80/image1.gif
//10.102.3.89:80/file5.html
//10.102.3.96:80/
//10.102.3.97:80/bg_logo_segue.gif
//10.102.3.89:80/file500.html
//10.102.3.92:80/
//10.102.3.96:80/cgi-bin/rfc/ccProxyReval.pl
Total URLs in IC = 10
```

The blank entries in the output are due to the default caching properties for GET / HTTP/1.1.

Flushing Objects

How can I flush a selective object from the cache?

You can identify an object uniquely by its complete URL. To flush such object, you can perform any of the following tasks:

- Flush cache
- Flush content group
- Flush the specific object

To flush the specific object, you have to specify the query parameters. You specify the invalParam parameter to flush the object. This parameter applies only to a query.

Does any change in the cache configuration trigger flushing of cache?

Yes. When you make any changes to the cache configuration, all the SET cache commands inherently flush the appropriate content groups.

I have updated the objects on the server. Do I need to flush the cached objects?

Yes. When you update objects on the server, you must flush the cached objects, or at least the relevant objects and content groups. The integrated cache is not affected by an update to the server. It continues to serve the cached objects until they expire.

Flash Cache

What is Flash Cache feature of the NetScaler appliance?

The phenomenon of Flash crowds occurs when a large number of clients access the same content. The result is a sudden surge in traffic toward the server. The Flash Cache feature enables the NetScaler appliance to improve performance in such situation by sending only one request to the server. All other requests are queued on the appliance and the single response is served to all of the requests. You can use either of the following commands to enable the Fast Cache feature:

- add cache contentGroup <Group_Name> -flashCache YES
- set cache contentGroup <Group_Name> -flashCache YES

What is the limit for Flash Cache clients?

The number of Flash Cache clients depends on the availability of resources on the NetScaler appliance.

Default Behaviour

Does the NetScaler appliance proactively receive objects upon expiry?

The NetScaler appliance never proactively receives objects on expiry. This is true even for the negative objects. The first access after expiry triggers a request to the server.

Does the integrated cache add clients to the queue for serving even before it starts receiving the response?

Yes. The integrated cache adds clients to the queue for serving even before it starts receiving the response.

What is the default value for the Verify cached object using parameter of the cache configuration?

HOSTNAME_AND_IP is the default value.

Does the NetScaler appliance create log entries in the log files?

Yes. The NetScaler appliance creates log entries in the log files.

Are compressed objects stored in the cache?

Yes. Compressed objects are stored in the cache.

Interoperability with other features

What happens to objects that are currently stored in cache and are being accessed though SSL VPN?

Objects stored in the cache and accessed regularly are served as cache hits when accessed through SSL VPN.

What happens to objects stored in the cache when accessed through SSL VPN and later accessed through a regular connection?

The objects stored through the SSL VPN access are served as a hit when accessed through the regular connection.

When using weblogging, how do I differentiate entries that indicate response served from cache from those served by the server?

For responses served from the integrated cache, the server log field contains the value IC. For responses served from a server, the server log field contains the value sent by the server. Following is a sample log entry for an integrated caching transaction:

"10.102.1.52 - "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 4.0; .NET CLR 1.0.3705)" "GET /" 200 0 "IC" 10.102.1.45"

Along with a client request, the response logged is the one sent to the client and not necessarily the one sent by the server.

Miscellaneous

What do you mean by configuring relexpiry and absexpiry?

By configuring relexpiry and absexpiry, it means that you are overriding the header irrespective of what appears in the header. You can configure different expiry setting and the content group level. With relexpiry, expiration of the header is based on the time at which the object was received by the NetScaler. With absexpiry, expiration is based on the time configured on the NetScaler. Relexpiry is configured in terms of seconds. Absexpiry is a time of day.

What do you mean by configuring weakpos and heuristic?

The weakpos and heuristic are like fall back values. If there is an expiry header, it is considered only if the last-modified header is present. The NetScaler appliance sets expiry on the basis of the last-modified header and the heuristic parameter. The heuristic expiry calculation determines the time to expiry by checking the last-modified header. Some percentage of the duration since the object was last modified is used as time to expiry. The heuristic of an object that remains unmodified for longer periods of time and is likely to have longer expiry periods. The –heurExpiryParam specifies what percentage value to use in this calculation. Otherwise, the appliance uses the weakpos value.

What should I consider before configuring dynamic caching?

If there is some parameter that is in name-value form and does not have the full URL query, or the appliance receives the parameter in a cookie header or POST body, consider configuring dynamic caching. To configure dynamic caching, you have to configure hitParams parameter.

How is hexadecimal encoding supported in the parameter names?

On the NetScaler appliance, the %HEXHEX encoding is supported in the parameter names. In the names that you specify for hitParams or invalParams, you can specify a name that contains %HEXHEX encoding in the names. For example, name, nam%65, and n%61m%65 are equivalent.

What is the process for selecting a hitParam parameter?

Consider the following excerpt of an HTTP header for a POST request:

```
How do we select a hitparam?
POST /data2html.asp?param1=value1&param2=&param3&param4=value4
HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/vnd.ms-powerpoint, application/vnd.ms-excel,
application/msword, application/x-shockwave-flash, */*
Referer: http://10.102.3.97/forms.html
Accept-Language: en-us
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Host: 10.102.3.97
Content-Length: 153
Connection: Keep-Alive
Cache-Control: no-cache
Cookie: ASPSESSIONIDQGQGGRNY=NNLLKDADEENOAFLCCDGFGDMO
S1=This+text+is+only+text%2C+not+more+and+not+less%2C+%0D%0Ajust+text+to+be+itself%2C+namely+%22Text%22+to+be+posted+as+text+%28what+else.
```

In the above request, you can use S1 and B1, highlighted in bold face for your reference, as hitParams depending on your requirements. Additionally, if you use -matchCookies YES in the ASPSESSIONIDQGQGGRNY content group, then you can also use these parameters as hitParams.

What happens to the queued clients if the response is not cacheable?

If the response is not cacheable, all of the clients in the queue receive the same response that the first client receives.

Can I enable the Poll every time (PET) and Flash Cache features on the same content group?

No. You cannot enable PET and Flash Cache on the same content group. The integrated cache does not perform AutoPET function on Flash Cache content groups. The PET feature ensures that the integrated cache does not serve a stored object without consulting the server. You can configure PET explicitly for a content group.

When are the log entries created for the queued clients?

The log entries are created for the queued clients soon after the appliance receives the response header. The log entries are created only if the response header does not make the object non-cacheable.

What is the meaning of the DNS, HOSTNAME, and HOSTNAME_AND_IP values of the Verify cached object using parameter of the cache configuration?

The meanings are as follows:

- set cache parameter -verifyUsing HOSTNAME
  This ignores the destination IP address.
- set cache parameter -verifyUsing HOSTNAME_AND_IP
  This matches the destination IP address.
- set cache parameter -verifyUsing DNS
  This uses the DNS server.

I have set weakNegRelExpiry to 600, which is 10 minutes. I noticed that 404 responses are not getting cached. What is the reason ?

This completely depends on your configuration. By default, 404 responses are cached for 10 minutes. If you want all 404 responses to be fetched form the server, specify–weakNegRelExpiry 0. You can fine tune the –weakNegRelExpiry to a desired value, such as higher or lower to get the 404 responses cached appropriately. If you have configured –absExpiry for positive responses, then it might not yield desired results.

When the user accesses the site by using the Mozilla Firefox browser, the updated content is served. However, when the user accesses the site by using the Microsoft Internet Explorer browser, stale content is served. What could be the reason?

The Microsoft Internet Explorer browser might be taking the content from its local cache instead of the NetScaler integrated cache. The reason could be that the Microsoft Internet Explorer browser is not respecting the expiry related header in the response.

To resolve this issue, you can disable the local cache of the Internet Explorer and clear the offline content. After clearing the offline content, the browser should display the updated content

What if Hits are zero?

Check to see if the server time and NS time are in sync. And the weakPosrelexpiry limit set should bear the time difference between NS and server as shown below

root@ns180# date

Tue May 15 18:53:52 IST 2012

Why are policies getting hits but nothing is being cached?

Verify that memory is allocated to the integrated cache and that the allocation is greater than zero.

Is it possible to zero the cache counters?

There is no command line or GUI option for setting the cache counters to zero, and flushing the cache does not do so either. Rebooting the box automatically sets these counters to zero.

Load Balancing

Feb 26, 2014

What are the various load balancing policies I can create on the NetScaler appliance?

You can create the following types of load balancing policies on the NetScaler appliance:

- Least Connections
- Round Robin
- Least response time
- Least bandwidth
- Least packets
- URL hashing
- Domain name hashing
- Source IP address hashing
- Destination IP address hashing
- Source IP - Destination IP hashing
- Token
- LRTM

Can I achieve the Web farm security by implementing load balancing using the NetScaler appliance?

Yes. You can achieve Web farm security by implementing load balancing using the NetScaler appliance. NetScaler appliance enables you to implement the following options of the load balancing feature:

- IP Address hiding: Enables you to install the actual servers to be on private IP address space for security reasons and for IP address conservation. This process is transparent to the end-user because the NetScaler appliance accepts requests on behalf of the server. While in the address hiding mode, the appliance completely isolates the two networks. Therefore, a client can access a service running on the private subnet, such as FTP or a Telnet server, through a different VIP on the appliance for that service.
- Port Mapping: Enables the actual TCP services to be hosted on non-standard ports for security reasons. This process is transparent to the end-user as the NetScaler appliance accepts requests on behalf of the server on the standard advertised IP address and port number.

What are various devices that I can use to load balance with a NetScaler appliance?

You can load balance following devices with a NetScaler appliance:

- Server farms
- Caches or Reverse Proxies
- Firewall devices
- Intrusion detection systems
- SSL offload devices
- Compression devices
- Content Inspection servers

Why should I implement the load balancing feature for the website?

You can implement the Load balancing feature for the website to take the following advantages:

- Reduce the response time: When you implement the load balancing feature for the website, one of the major benefits is the boost you can look forward to in load time. With two or more servers sharing the load of the web traffic, each of the servers runs less traffic load than a single server alone. This means there are more resources available to fulfill the client requests. This results in a faster website.

- Redundancy: Implementing the load balancing feature introduces a bit of redundancy. For example, if the website is balanced across three servers and one of them does not respond at all, the other two can keep running and the website visitors do not even notice any downtime. Any load balancing solution immediately stops sending traffic to the backend server that is not available.

Why do I need to disable the Mac Based Forwarding (MBF) option for Link Load Balancing (LLB)?

- If you enable the MBF option, the NetScaler appliance considers that the incoming traffic from the client and the outgoing traffic to the same client flow through the same upstream router. However, the LLB feature requires that the best path be chosen for the return traffic.
- Enabling the MBF option breaks this topology design by sending the outgoing traffic through the router that forwarded the incoming client traffic.

Load Balancing Methods

Feb 26, 2014

What are the various load balancing methods I can create on a NetScaler appliance?

You can create the following types of load balancing methods on a NetScaler appliance:

- Least Connections
- Round Robin
- Least response time
- Least bandwidth
- Least packets
- URL hashing
- Domain name hashing
- Source IP address hashing
- Destination IP address hashing
- Source IP - Destination IP hashing
- Token
- LRTM

What is the good practice I can follow when configuring the load balancing Virtual server on a NetScaler appliance?

The NetScaler appliance gives preference to the domain specified in the URL.

If domain appears in the URL as well as the HOST header, to what does the NetScaler appliance give preference in case of Domain Hashing?

The NetScaler appliance gives preference to the domain specified in the URL.

What health check for SMTP should I use?

You should use a TCP-ECV monitor with a send the QUIT string, which means "Ask the receiver to send a valid reply, and then close the transmission channel." You can leave the receive string blank, as it does not matter what the NetScaler appliance receive back. You can also use 250, which is a valid reply meaning "Requested mail action okay, completed".

You can use any of the following commands to use the TCP-ECV monitor to send the QUIT string:

- add monitor <name> TCP-ECV -send <string> -recv <string>
  add monitor smtp TCP-ECV –send “quit” –recv “”

- add monitor <name> TCP-ECV -send <string> -recv <string>
  add monitor smtp TCP-ECV –send “quit” –recv “250”

Why do I need to disable the Mac Based Forwarding (MBF) option for Link Load Balancing (LLB)?

If you enable the MBF option, the NetScaler appliance considers that the incoming traffic from the client and the outgoing traffic to the same client flow through the same upstream router. However, the LLB feature requires that the best path be chosen for the return traffic.

Enabling the MBF option breaks this topology design by sending the outgoing traffic through the router that forwarded the incoming client traffic.

The client is able to access the service directly. However, the client is not able to access the VIP address of

the NetScaler appliance. Have I made some incorrect configuration on the NetScaler appliance?

You might have enabled the Use Source IP (USIP) mode on the service bound to the virtual server on the NetScaler appliance. This mode forces the packets sent from the appliance to the backend server to contain the client IP as the source IP of the packet, whereas it should use the MIP. In this instance, probably the backend servers do not have a default route pointing to the appliance, which is preventing the server from sending the USIP packets back. However, it works when if you turn off the USIP mode for the services, as the servers can communicate with the MIP.

I want to configure the NetScaler appliance to support the persistence on a value. How can I achieve this configuration?

The issue with creating a rule for such a configuration is that there is nothing unique that you can use to persist. Therefore, you can create a wildcard rule or just match the jsession. In such cases, every request matches to any of the application servers. If each application server had a unique field in the jsession id, you can write a rule for persistence. Another option is to allow the NetScaler appliance to insert a cookie, such as jsessionID=<Value>.

Can I use the asterisk (*) wild card in the cookie, such as jsessionID=*, to represent a value the user has through the entire session?

No. The wildcard rule for persistence does not work with a dynamic session ID. In this case, all the users with a session ID persist to the server.

What are the various persistence types available on the NetScaler appliance?

The NetScaler appliance supports the following persistence types:

- Source IP
- Cookie insert
- SSL session ID
- URL passive
- Custom Server ID
- Rule
- DESTIP

Which counter is used in the Least Connection load balancing method?

When you configure the Least Connection load balancing method, the NetScaler appliance selects the server that has the least number of active transactions by using the Least Connection algorithm.

The Least Connection algorithm uses the Active Transactions (ATr) counter to implement the logic. You can run the following command from the shell prompt to display the details of the counter:

```
nsconmsg –s ConLb=2 –d oldconmsg
The following is the sample output of the command:
nsconmsg –s ConLb=2 –d oldconmsg
The following is the sample output of the command:
nsconmsg –s ConLb=2 –d oldconmsg
S(10.102.12.205:80:UP) Hits(2157, 0/sec, P[0, 0/sec]) ATr(0) Mbps(0.00) BWlmt(0
kbits) RspTime(0.00 ms)
Other: Pkt(1/sec, 0 bytes) Wt(10000) RHits(100)
Conn: CSvr(10, 0/sec) MCSvr(3) OE(1) RP(1) SQ(0)
```

The following is the list of counters used in the preceding output:

- ATr: Active Transactions. This is the number of active connections to the service.
- OE: Open Established. This is the number of connections to the service in the Established state.

- RP: Reuse Pool. This is the number of connections to the service stored in the reuse pool.
- SQ: Surge Queue. This is the number of connections to the service waiting in the surge queue

The Least Connection algorithm makes the following calculation before load balancing the requests:

- HTTP requests:
  ATr = OE – SQ - RP

  The ATr counter excludes the idle connections that are added to the reuse pool because these connections are reused to serve the new client requests.

- Non-HTTP Requests:
  ATr = OE – SQ

  The ATr counter includes all open connections because the idle connections are not added to the reuse pool.

What is a suitable deployment scenario for URL or Domain Hashing?

This form of load balancing method is more suitable for a cache environment where a cache serves a wide range of content from the Internet or the backend servers. By directing requests from a specific domain or URL to the cache that had previously served that domain, also known as hot cache, domain hashing makes sure a better resource utilization at the cache, higher cache-hit ratio, and faster request and response latency.

If domain appears in the URL as well as the HOST header, what does the NetScaler appliance give preference to in case of Domain Hashing?

The NetScaler appliance gives preference to the domain specified in the URL.

What happens if a NetScaler appliance cannot properly parse the HTTP request in case of URL Hashing?

If a NetScaler appliance cannot properly parse the HTTP request, such as if the request is in an unrecognized method or is a 0.9 HTTP request, the policy defaults to round-robin method for that request.

On a NetScaler appliance, what is considered as the Domain start and end for Domain Hashing?

On a NetScaler appliance, the Domain start is identified by the start of the domain string. The Domain end is the domain string without the port information.

On a NetScaler appliance, what is considered as the URL start and end for URL Hashing?

On a NetScaler appliance, the URL start is identified by the beginning of the /. The appliance skips the domain name and port. The URL end is the end of the GET sub-header.

What happens if a NetScaler appliance cannot parse the HTTP request properly in case of Domain Hashing?

If a NetScaler appliance cannot properly parse the HTTP request, such as if the request is in an unrecognized method or is a 0.9 HTTP request, or the request does not contain the host header, the policy defaults to round-robin method for that request.

If the actual URL length is smaller than the configured Hash length, how is hashing done in case of URL or Domain Hashing?

Hash function takes Minimum of hash length or actual URL or domain string length, whichever is smaller, for hash computation. The default value of hash length is 80 bytes, which is a user configurable parameter.

Is it possible configure the load balancing feature to make sure that the requests from the same subnet or requests to the same subnet are sent to the same server by using the Source IP, Destination IP, or Source IP

and Destination IP hashing method?

Yes. It possible configure the load balancing feature to make sure that the requests from the same subnet or requests to the same subnet are sent to the same server by using the Source IP, Destination IP, or Source IP and Destination IP hashing method.

By default, the NetScaler appliance uses the netmask 255.255.255.255. Therefore, the request is hashed for each Source IP, Destination IP, or Source IP and Destination IP addresses. You can use the -netmask <Netmask> option of either the add lb vserver or set lb vserver command to mask the Source IP, Destination IP, or Source IP and Destination IP addresses before calculating the hash value. This makes sure that all requests from the same subnet or requests to the same subnet are directed to the same server.

What is the deployment scenario I can use for the Source IP and Destination IP hashing method for load balancing?

You can use this load balancing method in IDS load balancing. The hashing is symmetric and yields the same value if the source IP and the destination IP addresses are reversed. This makes sure that all packets flowing from a specific client to the same destination are directed to the same IDS server.

What is the deployment scenario I can use for the Destination IP hashing method for load balancing?

This load balancing method is more appropriate when load balancing a transparent cache farm. This method is not recommended for the backend server and reverse proxy farm. This method is recommended in conjunction with cache redirection of transparent proxy farm.

Monitoring

Nov 07, 2016

Why should I configure a monitor?

Monitors help you in identifying the state of the service. In case a service is DOWN, monitor can identify it and the traffic is routed to alternative servers.

Can I bind multiple monitors to the same service?

Yes, you can bind multiple monitors to the same service. Each monitor evaluates the response to a different type of traffic.

I want to make some changes to built-in monitor. Can I do that?

No, you cannot customize the built-in monitors. You can only bind or unbind a built-in monitor to a service. Additionally, you cannot remove a built-in monitor.

I want to monitor SSL services. What monitors can I use?

You can use the following built-in monitors for an SSL service. You can also create your own monitors for SSL service.

- TCP
- HTTP
- TCP-ECV
- HTTP-ECV

I do not understand the difference between FTP monitor and FTP-EXTENDED monitors.

The FTP monitor checks the basic functionality. The FTP-EXTENDED monitor verifies whether the FTP server is able to transmit a file correctly.

I want to monitor the NetScaler Gateway. Is there a built-in monitor that I can use?

You can use the CITRIX-AG monitor for NetScaler Gateway.

What are custom monitors?

Custom monitors are based on the scripts that are included with the NetScaler operating system. Custom monitor can be of one of the following types:

- Inline monitor
- User monitor
- Load monitor

You can use the scripts provided with the NetScaler operating system or create your own scripts to monitor the services to which the monitors are bound.

Can I bind an inline monitor to Global Server Load Balancing (GSLB) remote or local service?

No, you cannot bind an inline monitor to GSLB remote or local service because these services represent the virtual servers rather than actual load balanced Web servers.

I have created a custom script for a user monitor, but I do not know where to add it.

You should add your script to /nsconfig/monitors directory.

Can I make changes to the scripts available in /nsconfig/monitors directory?

Yes, you can make changes to the scripts available in the /nsconfig/monitors directory and save it with a different name.

What happens when I remove all the monitors bound to the service?

When you remove all the monitors bound to the service, the default monitor is bound to the service. You cannot remove the default monitor bound to a service.

When should I use a reverse monitor? What purpose does it serve?

Reverse monitors are monitor that mark the service as DOWN when the probe criteria is met and marks the service as UP when criteria is not met.

Reverse monitors can be used in a situation when you want to use only one of the two available services. The reverse monitor marks the secondary service as DOWN as long as the primary service is UP. When primary service goes DOWN, it marks the secondary as UP. Reverse monitors monitor the server directly. These do not work as intended if you manually mark the service as DOWN.

What are the limitations for the Least Response Time Based on Monitoring load balancing method?

This load balancing method is not applicable to Global Server load balancing (GSLB) Virtual servers and session-less load balancing of type ANY. This load balancing is not advisable for HTTP and HTTPS type load balancing because there is LEASTRESPONSETIME load balancing, which considers live traffic to make decisions than the periodic monitoring probes.

What commands that I need to use to configure the Least Response Time Based on Monitoring load balancing method?

When you add services, make sure that appropriate monitors are bound to the service for LRTM to work. Later the load balancing method on the VIP must be set to LRTM. The following is a set of sample commands that you can run to configure LRTM on a NetScaler appliance:

```
add service svc1 10.102.4.66 ftp 21
add service svc2 10.102.4.67 ftp 21
add monitor con-ftp ftp –username nsroot –password <password>
bind monitor con-ftp svc1
bind monitor con-ftp svc2
add lb vserver ftp-vip ftp  10.1.1.1  21 –lbmethod LRTM
bind lb vserver ftp-vip svc1
bind lb vserver ftp-vip svc2
```

How frequently is the response time of a server measured?

The response time of the server is measured for every HTTP request.

Is the response time smoothed over a period?

The Least Response Time algorithm uses the average response time for the most recently completed 7-second polling interval. This provides some smoothing, but the algorithm does not strive for any greater complexity.

Is it possible to check the current response time value the NetScaler appliance is using for a server by using the command line interface?

Yes. You can display the real-time, time-to-first-byte (TTFB) response time metrics for all servers you have configured on a NetScaler appliance. You can display these on the Dashboard in either graphical or numeric values. Additionally, you can run the /etc/nsconmsg command to display these values from the command line interface. With the nsconmsg command, you can also display the historical values.

Is Average Response Time a good indicator for the server load?

The average response time might not be the most accurate form of estimating the server load because response time for

the dynamic content can be much higher than that of the static content, but on the assumption that the backend servers have replicated content, the average response time serves as a good approximation of the server load.

What health check for SMTP should I use?

You should use a TCP-ECV monitor with a send the QUIT string, which means "Ask the receiver to send a valid reply, and then close the transmission channel." You can leave the receive string blank, as it does not matter what the NetScaler appliance receive back. You can also use 250, which is a valid reply meaning "Requested mail action okay, completed".

You can use any of the following commands to use the TCP-ECV monitor to send the QUIT string:

- add monitor <name> TCP-ECV -send <string> -recv <string>
  add monitor smtp TCP-ECV –send "quit" –recv ""
- add monitor <name> TCP-ECV -send <string> -recv <string>
  add monitor smtp TCP-ECV –send "quit" –recv "250"

What is the difference between FTP monitor and FTP-EXTENDED monitors?

The FTP monitor checks the basic functionality. The FTP-EXTENDED monitor verifies whether the FTP server is able to transmit a file correctly.

For more information, see Monitors.

Persistence

Feb 26, 2014

I have configured Cookie Insert persistence on the NetScaler appliance. The users report that the connections work fine for some time, but then start getting disconnected. What best practice should I follow when configuring persistence?

By default, the time-out value for Cookie Insert persistence is 120 seconds. When you configure persistence for applications for which idle time cannot be determined, set the Cookie Insert persistence time-out value to 0. With this setting, the connection does not time out.

I have configured an HTTP virtual server on the NetScaler appliance. I need to make sure that a user always connects to the same server for the requested content, so I configured SourceIP persistence. Now, increasing the time-out value for persistence introduces latency. How can I increase the timeout value without affecting performance?

Consider using Cookie Insert persistence with the time-out value set to 0. This setting enables long-duration persistence settings, because the appliance does not specify a time for expiring the cookie.

The application server does not recognize cookies. Can I configure the NetScaler appliance to use Cookie Insert persistence?

On a NetScaler appliance, Cookie Insert persistence applies only to the client-side request, not to the application server. Additionally, when an application server does not set a cookie value, the response from the server does not contain any cookie value. However, the NetScaler appliance uses its cookie table to respond to the appropriate client.

I have configured Cookie Insert persistence on the NetScaler appliance. I works as expected when clients from the same time zone access the content. However, when a client from another time zone makes an attempt to connect, the connection is immediately timed out. What is the possible reason for this and how can I resolve this issue?

Time based Cookie Insert persistence works as expected when a client from the same time zone makes a connection. However, when the client machine and NetScaler appliance are in different time zones, the cookie is not valid. For example, when a client in EST time zone sends a cookie at 11:00 AM EST to a NetScaler appliance in the PST time zone, the appliance receives the cookie at 2:00 PM PST. As a result of the difference in time, the cookie is not valid, and the connection is immediately timed-out.
To resolve this issue, you can set the time-out value for Cookie Insert persistence to 0.

I have installed application servers, such as Oracle Weblogic server, that are load balanced by using a NetScaler appliance. To make sure that clients get persistent connections to these servers, I have configured SourceIP persistence. It works as expected when a connection is made from a computer. However, thin clients attempt a connection through a terminal server and, as a result, the appliance receives requests from multiple clients from the same IP address (the terminal server IP address). Therefore, the connections from all thin clients are directed to the same application server. Is it possible to configure persistency for requests from individual thin clients based on the client IP address?

No. The NetScaler appliance receives requests from the terminal server and the source IP address of the request remains the same. As a result, the appliance cannot distinguish among the requests received from the thin clients and provide persistence according to the requests from thin clients. To avoid this problem, you can configure Rule persistence based on some unique parameter value for each thin client.

I have multiple application servers load balanced through a NetScaler appliance. I have configured persistence along with monitors that monitor application servers every few seconds. If a server does not respond to a probe, the monitor marks the status of the server as DOWN. What happens to the client connections that are supposed to persist on the server that has just been marked as DOWN?

Regardless of persistence, the NetScaler appliance does not send traffic to a server that is marked as DOWN. The appliance overrides the persistence method and directs the traffic to another server that is available to receive the traffic according to the load balancing method configured on the appliance. The appliance renegotiates persistence for these connections.

I have configured a NetScaler appliance to load balance RDP connections. Even after configuring persistence for the RDP connections, users are connected to a different RDP server. Is there any method to make sure that the user is connected to the same server the next time?

No. This is an expected behavior. An RDP connection is a one-time connection. When a user disconnects an RDP connection and attempts to reconnect, the connection request is directed to an RDP server according to the load balancing method configured on the NetScaler appliance. The user might or might not get connected to the same server.

I have configured a NetScaler appliance to load balance application servers, such as Microsoft Exchange, with SourceIP persistency. The setup has a wireless network as a backup. When a user switches from the LAN to the wireless network, the connectivity to the applications is interrupted. For some applications, such as Microsoft Exchange, the user has to close the connection and reconnect to the application. Is there any solution for this issue?

No. This is an expected behavior. When you configure SourceIP persistence on the appliance, the persistence works according to the request from the source IP address. However, when you switch from LAN to the wireless network, the IP address of the client changes. As a result, persistence with the application server breaks.

I need to configure a NetScaler appliance to load balance HTTP application servers. For users connecting to these servers, I want to configure session-aware persistence. Which persistence method is recommended for this scenario?

Citrix recommends that you use the Cookie Insert persistence method for this scenario. When you configure this method, the session with the server persists until the client presents a valid cookie. To have the session persist for an indefinite length of time, set the cookie time-out value to 0, so that the cookie does not expire.

I have configured the NetScaler appliance to load balance Web Interface servers. When accessing the servers, the user receives the “State Error” error message. Additionally, when one of the Web Interface servers is shut down or not available, some of the users receive an error message. Are there any recommended settings that can resolve this issue?

Yes. Citrix recommends that you specify the Cookie Insert persistence method on the NetScaler appliance when load balancing Web Interface servers. Lack of persistence to the Web Interface servers can result in error messages when a user attempts to connect to the server.

I have configured multiple load balancing virtual servers for different functions of an application server. Is it possible to configure persistence with the same application server when a user accesses the server through any of the virtual servers?

Yes. You can create a persistency group with SourceIP persistence and bind the required virtual servers to the persistency group.

What is the data type of the CustomserverID parameter of the Custom Server ID persistence type?

Starting NetScaler release 10, the CustomserverID parameter is of alphanumeric data type. In the earlier releases, the parameter was of numeric data type.

What is the good practice I can follow when configuring the load balancing virtual server on a NetScaler appliance?

When configuring load balancing virtual servers, consider using Cookie Persistence option.

I want to configure the NetScaler appliance to support the persistence on a value. How can I achieve this configuration?

The issue with creating a rule for such a configuration is that there is nothing unique that you can use to persist. Therefore, you can create a wildcard rule or just match the jsession. In such cases, every request matches to any of the application servers. If each application server had a unique field in the jsession id, you can write a rule for persistence. Another option is to allow the NetScaler appliance to insert a cookie, such as jsessionID=<Value>.

Can I use the asterisk (*) wild card in the cookie, such as jsessionID=*, to represent a value the user has through the entire session?

No. The wildcard rule for persistence does not work with a dynamic session ID. In this case, all the users with a session ID persist to the server.

What are the various persistence types available on the NetScaler appliance?

The NetScaler appliance supports the following persistence types:

- Source IP
- Cookie insert
- SSL session ID
- URL passive
- Custom Server ID
- Rule
- DESTIP

See also:

To learn more about persistency groups, see Configuring Persistence Groups.

Protocols

Feb 26, 2014

What are the various protocols supported on the NetScaler appliance?

The NetScaler appliance supports the following protocols:

- HTTP
- FTP
- SSL
- SSL_BRIDGE
- SSL_TCP
- NNTP
- DNS
- DHCP Relay Agent (DHCPRA)
- Diameter
- MySQL
- MS SQL
- RADIUS
- RDP
- RTSP
- SIP

What is the difference among the SSL, SSL_BRIDGE, and SSL_TCP protocols?

The SSL, SSL_BRIDGE, and SSL_TCP protocols are used for the following tasks:

- SSL: The NetScaler appliance encrypts and decrypts data when using this protocol. It also has access to the underlying HTTP transaction.
- SSL_BRIDGE: The NetScaler appliance does not encrypt or decrypt any data when using this protocol. It does not have access to any HTTP data. The process of encryption and decryption is handled by the backend server.
- SSL_TCP: SSL_TCP implies TCP load balancing. When using this protocol, the NetScaler appliance encrypts and decrypts data but it does not have access to the underlying transaction.

How does the NetScaler appliance handle the load balancing of the HTTP or HTTPS traffic?

The HTTP or HTTPS load balancing is request based. If you have defined the service type as HTTP or HTTPS, the NetScaler appliance selects the server for every HTTP request independent of TCP connection. Therefore, different requests on the same client TCP connection can be Load Balanced to different servers on the backend.

How does the NetScaler appliance handle the load balancing of the TCP traffic?

The TCP load balancing is connection-based. If you have defined the service type as TCP, the NetScaler appliance selects the server for every new TCP connection. For each client connection, the appliance creates a connection to the backend server.

How does the NetScaler appliance handle the load balancing of the UDP traffic?

The UDP load balancing is time-based. If you have defined the service type as UDP, the NetScaler appliance selects the server for a UDP packet. After the server is selected, a session is created between the server and a client for a specific period of time. After the time expires, the session is deleted and new server selection is done for other packets even if the packets come from the same client.

Note: Session and load balancing are time based. Therefore, make sure that you specify the client timeout value. In case this value is set to 0, the appliance does not load balancing the same client requesting the same port.

How do I configure the TCP token based load balancing method?

You can set the additional properties of the VIP by using the set lb vserver command to configure TCP token based load balancing method. To configure this method with token received in the payload of the data packet at data offset 100 and length 2 bytes, run the following command:

set lb vserver <virtual servername> -lbmethod TOKEN –data_offset 100 –datalength 2

Note: The data offset and datalength are valid parameters only for TCP protocol.

How do I configure HTTP or HTTPS based token processing?

For HTTP protocol and token-based load balancing, a rule has to be maintained to check the string present in the URL. Depending on the string and token length, the token is extracted.

To create and configure an expression, use the add expression in the command. To define a rule, use the -rule <String> argument in the add lb vserver or set lb vserver commands. For example, the expression "URLQUERY contains token= -l 2" causes the NetScaler system to look for the token inside the URL query after matching the string token=, the length of the token is 2 bytes.

Service Groups

Feb 28, 2014

How can I evaluate the benefits of installing a NetScaler appliance on a network with respect to connection multiplexing?

Connection multiplexing is the method of reusing connections to avoid the overhead of establishing new TCP connections for data transfer when an established connection established is no longer in use. To evaluate the impact of a NetScaler appliance, you can run any of the following commands from the command line:

- stat service <Service_Name> detail
- stat servicegroup <Service_Group_Name> detail

The output of these commands displays the current client and server connections open on the appliance. The difference between the number of client connections and the number of server connections shows the extent to which the appliance reuses the same server connection for requests from multiple clients.

Is it possible to create a service group on the same port to which the application server listens?

Yes. If you create a service group on the same port to which the application server listens, the monitors for the services in the group mark the status of the services as DOWN if the port on the server is closed.

I have created a service group that takes care of the multiple servers. Is it possible to restrict internal access to some of the servers by using the same service group?

No. You cannot restrict internal access to some of the servers and provide external access to other servers by using the same service group. You can create another load balancing virtual server and bind the server targeted for internal access to this virtual server. The internal DNS server can resolve to this virtual server, and the external DNS server can resolve the other virtual server.

Alternatively, you can configure the content switching feature on the appliance. Depending on the client IP address of the request, you can switch traffic between the load balancing virtual servers that handle internal and external access to the servers.

Is it possible to identify IP address conflicts on a NetScaler appliance?

You can identify IP address conflicts from the Diagnostics page, which is available on the System node. On the Diagnostics page, click the View console messages link, and then click Run. The console messages display any IP address conflicts.

When the status of a service in a service group is marked as DOWN, the traffic of that service is diverted to other services. What happens to the traffic when the status of the service is later marked as UP?

When the service returns to the UP status, it handles new requests. However, the other services continue to handle all of their existing traffic until persistence expires.

I am installing a new NetScaler appliance. On this appliance, I would like to configure the same services as those on the existing appliance. Is it possible to export only the services configured on the existing appliance?

Yes. From the command line interface of the older appliance, open the ns.conf file, in the /nsconfig directory, and search to find the section that contains commands for services. Copy these commands to the ns.conf file on the new appliance.

Can I enable compression on a service group on which protocol ANY is enabled?

No. If you are using ANY as the protocol setting for a service group, the compression option is not available for the service

group. You need to select an appropriate protocol to enable compression.

Is it possible to create a secure load balancing virtual IP address that connects to a nonsecure service?

Yes. You can create a secure load balancing virtual IP address that can connect to a nonsecure service.

Can I bind a TCP service to a virtual server that uses the HTTPS protocol?

No. The underlying protocol of the service and the virtual server must be the same. For example, you can create an HTTPS virtual server to which you can bind an HTTP service. Similarly, you can create an SSL_TCP virtual server to which you can bind a TCP service.

I have a few service groups configured on the appliance. Each service group consists of multiple services. I want to direct a part of the traffic to one of the service groups. Is it possible to add weights to the service group to achieve this?

No. To direct a part of the traffic to a service group, you must add weights to each service in the service group.

For more information, see Configuring Services.

Migration

Oct 24, 2016

How is the rollback procedure performed on a NetScaler appliance?

The rollback procedure is similar to the basic upgrade procedure. Select the target build that you want to roll back to and perform the downgrade.
Before rolling back to a different release, Citrix recommends that you create a copy of your current configuration files. To downgrade from release 10.1, see Downgrading from Release 10.1 or to downgrade to an earlier build in 10.1, see Downgrading to an Earlier Build within Release 10.1 .

Can I upgrade a NetScaler appliance directly from release 9.3 to 10.1, or should I upgrade to 10.0 first and then to 10.1?

You can directly upgrade a NetScaler appliance regardless of the version or the build number.

Is the procedure for upgrading a NetScaler from classic to nCore different?

Yes. Only version 9.3 or earlier supports classic builds. For details, see Upgrading from a Classic to an nCore Release .

Does 10.0 support classic builds?

No. Release 10.0 supports only nCore builds. You must have a multi-processor Netscaler (MPX or higher) to upgrade to 10.0.

Can the primary appliance and secondary appliance have separate builds?

Recommended practice is to use the same version and build number on both the primary and the secondary appliance.

Can both the appliances in an High Availability (HA) pair be upgraded at the same time?

No. In an HA pair, first upgrade the secondary node and then upgrade the primary node. For details, refer Upgrading a High Availability Pair or Upgrading a NetScaler High Availability Pair to a Later Build.

Does Citrix support firmware upgrades in the amazon AWS cloud?

Yes.

Can I upgrade the NetScaler instance independently of the SDX version.

It is not required to upgrade the SDX version when the NetScaler appliance is upgraded. However, some features might not work.

Can I use the FTP server to upgrade the NetScaler appliance?

No. You must first download the firmware from the Citrix download site, save it on your local computer and then upgrade the appliance.

Is the procedure for upgrading the NetScaler appliance with GSLB configurations different from an upgrade of an appliance that is not involved in GSLB?

No. The upgrade procedure is similar to the basic upgrade procedure. The only difference is that you can upgrade the standalone or HA appliances on different sites in a phased manner.

SDX

Sep 04, 2013

What is SDX?

SDX is a true service delivery networking platform for enterprises and cloud datacenters. SDX features an advanced virtualization architecture that supports multiple NetScaler instances on a single hardware appliance.

When do I need SDX?

If you have multiple enterprise applications that have independent life cycle needs for L4–L7 networking services, or if you have a need to consolidate multiple underutilized load balancing appliances, you benefit from SDX.

What's unique about SDX?

SDX uniquely delivers key benefits from advancements in server hardware virtualization, hardware-assisted SSL acceleration, and the market-proven, award-winning NetScaler product line. The Management Service features an advanced control plane to unify provisioning, monitoring, and management in the most demanding multitenant environments, while providing full resource isolation for data separation and to meet service level agreement guarantees, such as availability, reliability, and performance.

How will I benefit from SDX?

SDX delivers isolated multitenancy with up to 40:1 consolidation. As a key pillar in Citrix's TriScale technology framework, SDX addresses the growing need to "scale in" within virtual data centers and cloud network infrastructures. The TriScale scale-in factor enables IT to provide the foundation for consolidating L4–L7 network services today, thereby simplifying the build-out of cloud based services down the line, in accordance with business requirements.

Will I need to go outside my normal procurement procedure to purchase SDX?

SDX is a fully contained networking appliance, designed for network deployment. SDX is not designed to be managed through standard hypervisor management tools such as XenCenter.

How do I purchase an SDX?

An SDX order has three basic product components: SDX appliance SKU, SDX support contract SKU, and Add-On Instance Packs. SKUs are also available for platform conversion (MPX-to-SDX) and platform upgrade (SDX-to-SDX). SDX today is available in Platinum Edition only.

Is there SDX-specific documentation?

Yes, please visit http://support.citrix.com/proddocs/topic/netscaler/sdx-ag-wrapper-con.html.

Do NetScaler editions apply to NetScaler SDX?

The editions do not apply from a packaging perspective. NetScaler SDX appliances and the instance 5-packs are priced the same regardless of the edition. However, when provisioning new instances, the administrator is free to deploy the Standard, Enterprise, or Platinum edition of the NetScaler software.

How much memory can I assign to each instance?

There is no maximum limit to the memory that can be assigned to each instance. Minimum memory required per instance is 2GB.

Can we migrate the existing configuration (ns.conf) from the MPX platform to SDX VPX instance?

Yes, but some configuration, such as RBA policies and SNMP community configuration, is deleted.

What NetScaler features do I get with SDX?

All NetScaler features are available on SDX.

Does SDX accelerate SSL in hardware like MPX does?

Yes. You can assign SSL cores to an instance during provisioning.

What changes to my network are required for me to deploy SDX?

SDX fits into your network environment through standard Ethernet interfaces. You must disable link aggregation control protocol (LACP) on any external switch ports connected to the appliance.

Is SDX interoperable with my routing and switching infrastructure?

Yes, although link aggregation control protocol (LACP) is currently not supported. However, SDX supports manual link aggregation.

Is SDX interoperable with my existing NetScaler deployment?

Yes, although standard VPX-to-MPX limitations apply. For example, high availability is supported only across homogeneous devices (you cannot pair a virtual device with a physical device), some configuration, such as RBA policies and SNMP configuration, is deleted, and license transfer is not supported.

Can I manage SDX from Command Center?

Yes. You can identify SDX appliances and provision and de-provision VPX instances by using Command Center.

How does SDX deliver multitenancy?

Each instance runs as a separate virtual machine with its own dedicated NetScaler kernel, CPU resources, memory resources, address space, and bandwidth allocation. Network I/O is done in a way that not only maintains aggregate system performance but also enables complete segregation of each tenant's data-plane and management-plane traffic.

Do I need to manage an SDX through XenCenter?

No. XenCenter is not supported. Use the Management Service to manage XenServer.

We are a VMware shop. We have no infrastructure available to support XenServer, do you have a VMware variant of SDX?

No additional XenServer infrastructure is necessary. SDX is a fully contained networking appliance with its own control plane, and the virtualization layer is transparent to the deployment.

Why is the system health monitoring page not showing any data?

You have to install the supplemental pack before you can use this feature. For installation instructions for the supplemental pack, see http://support.citrix.com/article/CTX132877.

How do I verify that the supplemental pack installation was successful?

After installation, a pop up window shows whether installation was successful or if there was an error.

Why is the VPX instance not reachable after interfaces on the appliance are modified?

When you provision a NetScaler VPX instance with L2VLAN configuration, physical interfaces on the SDX appliance are mapped to virtual interfaces on the VPX instance. If you remove an interface, you might change the mapping between the physical interfaces and VPX instances, and therefore you might lose connectivity to the VPX instance.
For example,

1. You provision a VPX instance, by using the Management Service, with interfaces 10/1, 10/2, 10/7, and tag VLAN 512 to interface 10/2. When you log on to that VPX instance, you see that interfaces 10/1, 10/2, and 10/3 are configured.
2. If you later modify the instance and remove interface 10/1, you lose connectivity to the instance, because interface 10/2 is renamed to 10/1 in the VPX instance.

Are IPv6 addresses supported on the NetScaler SDX appliance?

Yes. All NetScaler-supported IPv6 functionality is available on the SDX appliance.

Where are link parameters, such as speed and duplex, configured?

Link parameters are configured from the Management Service.

Should the appliance be restarted if the platform license is upgraded?

No. You do not need to restart the appliance for the new license to apply.

Do I need to restart the appliance to upgrade the device-level firmware?

Yes, this upgrade is handled through the Management Service and requires that the appliance be restarted. This is the only time that the SDX appliance needs a complete restart.

Do I need to restart the appliance when I upgrade it by using a Pay-As-You-Grow license?

No. Upgrading the appliance upgrades the platform license. Restart the Management Service but not the instances running on the SDX appliance. Once upgraded, the Management Service detects the higher throughput available for the instances. If you decide to increase the bandwidth limit for an instance, restart that instance after modifying the bandwidth limit.

What happens to production instances if I remove my platform license?

There is no change to the production instances. However, you cannot add new instances.

How can we readd a gadget to the Home page?

Click the << button in the top-right corner of the Home page. Then, type the name of the gadget, or press Enter for all gadgets. Click "Add to Dashboard".

Should member interfaces in manual link aggregation be part of same VLAN?

Yes. Member interfaces in manual link aggregation should be part of the same VLAN.

How many VLANs are supported per interface with VLAN filtering enabled? What happens if I configure more?

With VLAN filtering enabled, 10G interfaces support up to 63 VLANs, and 1G interfaces suppport up to 31 VLANs. This is a hard limit based on the number of the queues supported by the NIC. An error message appears if the limit is exceeded.

How many instances can be shared on a single NIC?

For a 10G interface, SDX supports up to 63 virtual functions per physical port, which translates to 63 instances per 10G NIC. For 1G interfaces, the maximum number of shared instances per NIC is 7.

Why is the XenServer password the same as the Management Service password?

The XenServer password and the Management Service password are the same to maintain administrative consistency.

Changing the XenServer password causes the internal communication between the Management Service and XenServer to fail.

If I have separate management networks, do I need to manually add these networks to the Management Service?

No. Communication is over an external device.

Why can't I modify the default administrator profile?

The default administrator profile enables multiple administrative roles to exist on the SDX. You cannot change the password of the nsroot administrator profile, but you can create a new administrator profile and make it the default profile.

Why does Core usage show 50% when I'm not passing any traffic through my NetScaler instance?

CPU core usage shows, from the hypervisor perspective, the CPU utilization of one physical CPU, which has two hyperthreads: one for the packet engine and one for the management CPU. For example, assume a single instance with one dedicated core. Even if you are not passing any traffic through your appliance, PE CPU utilization will be 100%, and average core utilization will be 50%.

Will restarting the Management Service interrupt my production instances?

No. Your production instances will continue to pass traffic without interruption while the Management Service restarts. The same applies when you upgrade the Management Service.

Can I configure the Management Service to send syslog?

Syslog through the Management Service is currently not supported.

Am I required to upgrade all VPX instances if I upgrade the Management Service?

No, instance life cycles can be managed independently of one other and of the life cycle of the Management Service.

If my Management Service and VPX instances are on different networks, how can I manage the VPX instance through HTTPS?

The same way as if they are on the same network.

If my Management Service and VPX instances are on different networks, how can I manage the VPX instance through the Management Service?

If the Management Service and the VPX instance are in different networks but the instance can be reached from Management Service, the Management Service shows the instance as UP. If an instance is UP, you can manage it from the Management Service. However, if communication between the two fails, the Management Service shows the instance as "Out of Service".

I forgot the IP address of my Management Service. What can I do?

Log on to XenServer, and then use the default IP address (169.254.0.10) to log on to the Management Service. At the shell prompt, type networkconfig to view or modify the IP address of the Management Service.

Can I specify VLANs on management interfaces?

VLANs on management interfaces are currently not supported.

How do I restart XenServer?

The only supported method for restarting XenServer is from the Management Service. It is equivalent to restarting the appliance.

How many instances can I provision on the SDX appliance? How much aggregate throughput can I expect?

This number is dependent on the hardware and the license that you purchased, as shown below:

- 11500, 13500, 14500, 16500, 18500, 20500—5 to 20 instances. Throughput ranges from 8 to 42 Gbps.
- 17500, 19500, 21500—5 to 20 instances. Throughput ranges from 20 to 50 Gbps.
- 17550, 19550, 20550, 21550—5 to 40 instances. Throughput ranges from 20 to 50 Gbps.
- 8400, 8600—2 to 5 instances. Throughput ranges from 4 to 6 Gbps.

Note: For more information, see the NetScaler datasheet at http://www.citrix.com/content/dam/citrix/en_us/documents/products/netscaler-data-sheet.pdf

Can I restrict functionality on the VPX instances?

Some functionality can be restricted by specifying the license (Standard, Enterprise, or Platinum) when you provision the instance.

How many SDX models are there, and how do they differ?

The NetScaler SDX appliance comes in the following variants:

- SDX 11500/13500/14500/16500/18500/20500—8 to 42 Gbps, maximum 20 instances, 8x1G ports, 4x10G ports.
- SDX 17500/19500/21500—20 to 50 Gbps, maximum 20 instances, 8x10G ports.
  Note: This platform is going EOS this year.
- SDX 17550/19550/20550/ 21550—20 to 50 Gbps, maximum 40 instances, 8x10G ports.
- SDX 8400/8600—4 to 6 Gbps, maximum 5 instances, (6x10/100/1000Base-T copper Ethernet ports + 6x1G SFP) and (6x10/100/1000Base-T copper Ethernet ports + 2x10G SFP+)

What is the minimum NetScaler software version required for SDX instances?

NetScaler VPX instances should run release 9.3 and later to be able to work on SDX.

How many physical interfaces will I need to use?

If you have a single management network, you'll need on an average 1 or 2 physical NICs per instance. For 2 or more management networks (multiple VLANs for NetScaler IP addresses), you'll need a dedicated separate physical NIC for each management VLAN trunk. You can share physical NICs among multiple instances with L2 separation. Therefore, depending on your topology, you can offset the management VLAN trunk count with multiple instances sharing a physical NIC.

Can I upgrade my MPX to an SDX? What about my MPX FIPS platform?

A non-FIPS MPX platform that supports the SDX architecture can be converted to a similar class of SDX platform. The MPX platform must have a platinum license to be eligible for this upgrade. This is a one way upgrade, and it wipes out the entire configuration on that MPX platform. For more information about this upgrade, see http://support.citrix.com/article/CTX129423.

How many SSL cards (cores) are supported on a NetScaler SDX appliance?

The number of SSL cards supported varies by the platform as follows:

- SDX 17500/19500/21500—16 cards.
- SDX 11500/13500/14500/16500/18500/20500—16 cards.
- SDX 17550/19550/20550/21550—36 cards.
- SDX 8400/8600—4 cards.

Note: Instances cannot share SSL cores. Any SSL cores that are allocated at the time of provisioning an instance are

dedicated to that instance.

Can I apply my VPX license to SDX?

No. NetScaler SDX and NetScaler VPX have different licensing models. One license cannot be used for the other.

Why are the hardware sensors not displayed on the NetScaler SDX 17500/19500/21500 appliance?

The NetScaler SDX 17500/19500/21500 is built on the MPX 17500/19500/21500 hardware platform. These appliance configurations do not support monitoring of hardware components.

When I upgraded my MPX to an SDX, the LCD panel went dark. Is that expected?

Yes, that is normal behavior. SDX does not support the LCD panel.

What are RX and TX errors on the NetScaler SDX appliance?

RX and TX errors include cyclic redundancy check (CRC) errors and small or runt packet errors.

What happens if a hardware component is removed from the SDX appliance?

If a hardware component is physically removed from the appliance, it no longer appears in the Management Service user interface.

Do I need to restart my appliance after I reconfigure VLAN filtering?

No. However, you need to restart the VPX instances that are affected by this change. The Management Service restarts the affected instances if you select "Reboot associated Instances" in the Enable/Disable VLAN Filter dialog box.

What is the NMI button for on the SDX appliance?

The NMI button is not operational on the SDX appliance.

SSL

Jun 16, 2014

HTTPS access to the NetScaler configuration utility fails on a VPX instance. How do I gain access?

A certificate-key pair is required for HTTPS access to the NetScaler configuration utility. On a NetScaler ADC, a certificate-key pair is automatically bound to the internal services. On an MPX or SDX appliance, the default key size is 1024 bytes, and on a VPX instance, the default key size is 512 bytes. However, most browsers today do not accept a key that is less than 1024 bytes. As a result, HTTPS access to the VPX configuration utility is blocked.

Citrix recommends that you install a certificate-key pair of at least 1024 bytes and bind it to the internal service for HTTPS access to the configuration utility or update the ns-server-certificate to 1024 bytes. You can use HTTP access to the configuration utility or the NetScaler command line to install the certificate.

If I add a license to an MPX appliance, the certificate-key pair binding is lost. How do I resolve this problem?

If a license is not present on an MPX appliance when it starts, and you add a license later and restart the appliance, you might lose the certificate binding. You must reinstall the certificate and bind it to the internal service

Citrix recommends that you install an appropriate license before starting the appliance.

What are the various steps involved in setting up a secure channel for an SSL transaction?

Setting up a secure channel for an SSL transaction involves the following steps:

1. The client sends an HTTPS request for a secure channel to the server.
2. After selecting the protocol and cipher, the server sends its certificate to the client.
3. The client checks the authenticity of the server certificate.
4. If any of the checks fail, the client displays the corresponding feedback.
5. If the checks pass or the client decides to continue even if a check fails, the client creates a temporary, disposable key called the *pre-master secret* and encrypts it by using the public key of the server certificate.
6. The server, upon receiving the pre-master secret, decrypts it by using the server's private key and generates the session keys. The client also generates the session keys from the pre-master secret. Thus both client and server now have a common session key, which is used for encryption and decryption of application data.

I understand that SSL is a CPU-intensive process. What is the CPU cost associated with the SSL process?

The following two stages are associated with the SSL process:

- The initial handshake and secure channel setup by using the public and private key technology.
- Bulk data encryption by using the asymmetric key technology.

Both of the preceding stages can affect server performance, and they require intensive CPU processing for of the following reasons:

1. The initial handshake involves public-private key cryptography, which is very CPU intensive because of large key sizes (1024bit, 2048bit, 4096bit).
2. Encryption/decryption of data is also computationally expensive, depending on the amount of data that needs to be encrypted or decrypted.

What are the various entities of an SSL configuration?

An SSL configuration has the following entities:

- Server certificate
- Certificate Authority (CA) certificate

- Cipher suite that specifies the protocols for the following tasks:
  - Initial key exchange
  - Server and client authentication
  - Bulk encryption algorithm
  - Message authentication
- Client authentication
- CRL
- SSL Certificate Key Generation Tool that enables you to create the following files:
  - Certificate request
  - Self signed certificate
  - RSA and DSA keys
  - DH parameters

I want to use the SSL offloading feature of the Citrix NetScaler appliance. What are the various options for receiving an SSL certificate?

You must receive an SSL certificate before you can configure the SSL setup on the Citrix NetScaler appliance. You can use any of the following methods to receive an SSL certificate:

- Request a certificate from an authorized CA.
- Use the existing server certificate.
- Create a certificate-key pair on the Citrix NetScaler appliance.
  Note: This is a test certificate signed by the test Root-CA generated by the NetScaler. Test certificates signed by this Root-CA are not accepted by browsers. The browser throws a warning message stating that the server's certificate cannot be authenticated.
- For anything other than test purposes, you must provide a valid CA certificate and CA key to sign the server certificate.

What are the minimum requirements for an SSL setup?

The minimum requirements for configuring an SSL setup are as follows:

- Obtain the certificates and keys.
- Create a load balancing SSL virtual server.
- Bind HTTP or SSL services to the SSL virtual server.
- Bind certificate-key pair to the SSL virtual server.

What are the limits for the various components of SSL?

SSL components have the following limits:

- Bit size of SSL certificates: 4096.
- Number of SSL certificates: Depends on the available memory on the appliance.
- Maximum linked intermediate CA SSL certificates: 9 per chain.
- CRL revocations: Depends on the available memory on the appliance.

What are the various steps involved in the end-to-end data encryption on a Citrix NetScaler appliance?

The steps involved in the server-side encryption process on a Citrix NetScaler appliance are as follows:

1. The client connects to the SSL VIP configured on the Citrix NetScaler appliance at the secure site.
2. After receiving the secure request, the appliance decrypts the request, applies layer 4-7 content switching techniques and load balancing policies, and selects the best available backend Web server for the request.
3. The Citrix NetScaler appliance creates an SSL session with the selected server.
4. After establishing the SSL session, the appliance encrypts the client request and sends it to the Web server by using the secure SSL session.
5. When the appliance receives the encrypted response from the server, it decrypts and re-encrypts the data, and sends the

data to the client by using the client side SSL session.

The multiplexing technique of the Citrix NetScaler appliance enables the appliance to reuse SSL sessions that have been established with the Web servers. Therefore, the appliance avoids the CPU intensive key exchange, known as *full handshake*. This process reduces the overall number of SSL sessions on the server and maintains end-to-end security.

Can I place the certificate and key files at any location? Is there any recommended location to store these files?

You can store the certificate and key files on the Citrix NetScaler appliance or a local computer. However, Citrix recommends that you store the certificate and key files in the /nsconfig/ssl directory of the Citrix NetScaler appliance. The /etc directory exists in the flash memory of the Citrix NetScaler appliance. This provides portability and facilitates backup and restoration of the certificate files on the appliance.
Note: Make sure that the certificate and the key files are stored in the same directory.

What is the maximum size of the certificate key supported on the Citrix NetScaler appliance?

A Citrix NetScaler appliance running a software release earlier than release 9.0 supports a maximum certificate key size of 2048 bits. Release 9.0 and later support a maximum certificate key size of 4096 bits. This limit is applicable to both RSA and DSA certificates.
An MPX appliance supports certificates from 512-bits up to the following sizes:

- 4096-bit server certificate on the virtual server
- 4096-bit client certificate on the service
- 4096-bit CA certificate (includes intermediate and root certificates)
- 4096-bit certificate on the back end server
- 4096-bit client certificate (if client authentication is enabled on the virtual server)

A virtual appliance supports certificates from 512-bits up to the following sizes:

- 4096-bit server certificate on the virtual server
- 4096-bit client certificate on the service
- 4096-bit CA certificate (includes intermediate and root certificates)
- 2048-bit certificate on the back end server
- 2048-bit client certificate (if client authentication is enabled on the virtual server)

What is the maximum size of the DH parameter supported on the Citrix NetScaler appliance?

The Citrix NetScaler appliance supports a DH parameter of maximum 2048 bits.

What is the maximum certificate-chain length, that is, the maximum number of certificates in a chain, supported on a Citrix NetScaler appliance?

A Citrix NetScaler appliance can send a maximum of 10 certificates in a chain when sending a server certificate message. A chain of the maximum length includes the server certificate and nine intermediate CA certificates.

What are the various certificate and key formats supported on the Citrix NetScaler appliance?

The Citrix NetScaler appliance supports the following certificate and key formats:

- Privacy Enhanced Mail (PEM)
- Distinguished Encoding Rule (DER)

Is there a limit for the number of certificates and keys that I can install on the Citrix NetScaler appliance?

No. The number of certificates and keys that can be installed is limited only by the available memory on the Citrix NetScaler appliance.

I have saved the certificate and key files on the local computer. I want to transfer these files to the Citrix NetScaler appliance by using the FTP protocol. Is there any preferred mode for transfering these files to the Citrix NetScaler appliance?

Yes. If using the FTP protocol, you should use binary mode to transfer the certificate and key files to the Citrix NetScaler appliance.
Note: By default, FTP is disabled. Citrix recommends using the SCP protocol for transferring certificate and key files. The configuration utility implicitly uses SCP to connect to the appliance.

What is the default directory path for the certificate and key?

The default directory path for the certificate and key is /nsconfig/ssl.

When adding a certificate and key pair, what happens if I do not specify an absolute path to the certificate and key files?

When adding a certificate and key pair, if you do not specify an absolute path to the certificate and key files, the Citrix NetScaler appliance searches the default directory, /nsconfig/ssl, for the specified files and attempts to load them to the kernel. For example, if the cert1024.pem and rsa1024.pem files are available in the /nsconfig/ssl directory of the appliance, both of the following commands are successful:

```
add ssl certKey cert1 -cert cert1204.pem -key rsa1024.pem

add ssl certKey cert1 -cert /nsconfig/ssl/cert1204.pem -key /nsconfig/ssl/rsa1024.pem
```

I have configured a high availability setup. I want to implement the SSL feature on the setup. How should I handle the certificate and key files in a high availability setup?

In a high availability setup, you must store the certificate and key files on both the primary and the secondary Citrix NetScaler appliance. The directory path for the certificate and key files must be the same on both appliances before you add an SSL certificate-key pair on the primary appliance.

What is a NULL-Cipher?

Ciphers with no encryption are known as NULL-Ciphers. For example, NULL-MD5 is a NULL-Cipher.

Are the NULL-Ciphers enabled by default for an SSL VIP or an SSL service?

No. NULL-Ciphers are not enabled by default for an SSL VIP or an SSL service.

What is the procedure to remove NULL-Ciphers?

To remove the NULL-Ciphers from an SSL VIP, run the following command:
bind ssl cipher <SSL_VIP> REM NULL

To remove the NULL-Ciphers from an SSL Service, run the following command:

```
bind ssl cipher <SSL_Service> REM NULL -service
```

What are the various cipher aliases supported on the Citrix NetScaler appliance?

The Citrix NetScaler appliance supports the following cipher aliases:

1. Alias Name: ALL
   Description: All NetScaler-supported ciphers, excluding NULL ciphers

2. Alias Name: DEFAULT
   Description: Default cipher list with encryption strength >= 128bit

3. Alias Name: kRSA
   Description: Ciphers with RSA key exchange algorithm
4. Alias Name: kEDH
   Description: Ciphers with Ephemeral-DH key exchange algorithm
5. Alias Name: DH
   Description: Ciphers with DH key exchange algorithm
6. Alias Name: EDH
   Description: Ciphers with DH key exchange algorithm and authentication algorithm
7. Alias Name: aRSA
   Description: Ciphers with RSA authentication algorithm
8. Alias Name: aDSS
   Description: Ciphers with DSS authentication algorithm
9. Alias Name: aNULL
   Description: Ciphers with NULL authentication algorithm
10. Alias Name: DSS
    Description: Ciphers with DSS authentication algorithm
11. Alias Name: DES
    Description: Ciphers with DES encryption algorithm
12. Alias Name: 3DES
    Description: Ciphers with 3DES encryption algorithm
13. Alias Name: RC4
    Description: Ciphers with RC4 encryption algorithm
14. Alias Name: RC2
    Description: Ciphers with RC2 encryption algorithm
15. Alias Name: eNULL
    Description: Ciphers with NULL encryption algorithm
16. Alias Name: MD5
    Description: Ciphers with MD5 message authentication code (MAC) algorithm
17. Alias Name: SHA1
    Description: Ciphers with SHA-1 MAC algorithm
18. Alias Name: SHA
    Description: Ciphers with SHA MAC algorithm
19. Alias Name: NULL
    Description: Ciphers with NULL encryption algorithm
20. Alias Name: RSA
    Description: Ciphers with RSA key exchange algorithm and authentication algorithm

21. Alias Name: ADH
    Description: Ciphers with DH key exchange algorithm, and NULL authentication algorithm
22. Alias Name: SSLv2
    Description: SSLv2 protocol ciphers
23. Alias Name: SSLv3
    Description: SSLv3 protocol ciphers
24. Alias Name: TLSv1
    Description: SSLv3/TLSv1 protocol ciphers
25. Alias Name: TLSv1_ONLY
    Description: TLSv1 protocol ciphers
26. Alias Name: EXP
    Description: Export ciphers
27. Alias Name: EXPORT
    Description: Export ciphers
28. Alias Name: EXPORT40
    Description: Export ciphers with 40-bit encryption
29. Alias Name: EXPORT56
    Description: Export ciphers with 56-bit encryption
30. Alias Name: LOW
    Description: Low strength ciphers (56-bit encryption)
31. Alias Name: MEDIUM
    Description: Medium strength ciphers (128-bit encryption)
32. Alias Name: HIGH
    Description: High strength ciphers (168-bit encryption)
33. Alias Name: AES
    Description: AES ciphers
34. Alias Name: FIPS
    Description: FIPS-approved ciphers
35. Alias Name: ECDHE
    Description: Elliptic Curve Ephemeral DH Ciphers

What is the command to display all the predefined ciphers of the Citrix NetScaler appliance?

To display all the predefined ciphers of the Citrix NetScaler appliance, at the NetScaler command line, type:
show ssl cipher

What is the command to display the details of an individual cipher of the Citrix NetScaler appliance?

To display the details of an individual cipher of the Citrix NetScaler appliance, at the NetScaler command line, type:
show ssl cipher <Cipher_Name/Cipher_Alias_Name/Cipher_Group_Name>

Example:

```
 > show cipher SSL3-RC4-SHA
 1) Cipher Name: SSL3-RC4-SHA
 Description: SSLv3 Kx=RSA Au=RSA Enc=RC4(128)
Mac=SHA1
 Done
```

What is the significance of adding the predefined ciphers of the Citrix NetScaler appliance?

Adding the predefined ciphers of the Citrix NetScaler appliance causes the NULL-Ciphers to get added to an SSL VIP or an SSL service.

Why do I need to bind the server certificate?

Binding the server certificates is the basic requirement for enabling the SSL configuration to process SSL transactions.
To bind the server certificate to an SSL VIP, at the NetScaler command line, type:

bind ssl vserver <vServerName> -certkeyName <cert_name>

To bind the server certificate to an SSL service, at the NetScaler command line, type:

bind ssl service <serviceName> -certkeyName <cert_name>

How many certificates can I bind to an SSL VIP or an SSL service?

On a NetScaler virtual appliance, you can bind a maximum of two certificates to an SSL VIP or an SSL service, one each of type RSA and type DSA. On a NetScaler MPX or MPX-FIPS appliance, if SNI is enabled, you can bind multiple server certificates of type RSA. If SNI is disabled, you can bind a maximum of one certificate of type RSA.
Note: DSA certificates are not supported on MPX or MPX-FIPS platforms.

Does SNI support Subject Alternative Name (SAN) certificates?

No. On a NetScaler appliance, SNI is not supported with a SAN extension certificate.

What happens if I unbind or overwrite a server certificate?

When you unbind or overwrite a server certificate, all the connections and SSL sessions created by using the existing certificate are terminated. When you overwrite an existing certificate, the following message appears:
ERROR:

Warning: Current certificate replaces the previous binding.

How do I install an intermediate certificate on Citrix NetScaler and link to a server certificate?

See the article at http://support.citrix.com/article/ctx114146 for information about installing an intermediate certificate.

Why am I am getting a "resource already exists" error when I try to install a certificate on the Citrix NetScaler?

See the article at http://support.citrix.com/article/CTX117284 for instructions for resolving the "resource already exists" error.

I want to create a server certificate on a Citrix NetScaler appliance to test and evaluate the product. What is the procedure to create a server certificate?

Perform the following procedure to create a test certificate.
Note: A certificate created with this procedure cannot be used to authenticate all the users and browsers. After using the certificate for testing, you should obtain a server certificate signed by an authorized Root CA.

To create a self-signed server certificate:

1. To create a Root CA certificate, at the NetScaler command line, type:
   create ssl rsakey /nsconfig/ssl/test-ca.key 1024

   create ssl certreq /nsconfig/ssl/test-ca.csr -keyfile /nsconfig/ssl/test-ca.key

   Enter the required information when prompted, and then type the following command:

   create ssl cert /nsconfig/ssl/test-ca.cer /nsconfig/ssl/test-ca.csr ROOT_CERT -keyfile /nsconfig/ssl/test-ca.key

2. Perform the following procedure to create a server certificate and sign it with the root CA certificate that you just created
   1. To create the request and the key, at the NetScaler command line, type:
      create ssl rsakey /nsconfig/ssl/test-server.key 1024

      create ssl certreq /nsconfig/ssl/test-server.csr -keyfile /nsconfig/ssl/test-server.key

   2. Enter the required information when prompted.
   3. To create a serial-number file, at the NetScaler command line, type:
      shell
      # echo '01' >
      /nsconfig/ssl/serial.txt
      # exit
3. To create a server certificate signed by the root CA certificate created in step 1, at the NetScaler command line, type:
   create ssl cert /nsconfig/ssl/test-server.cer /nsconfig/ssl/test-server.csr SRVR_CERT -CAcert /nsconfig/ssl/test-ca.cer -CAkey /nsconfig/ssl/test-ca.key -CAserial /nsconfig/ssl/serial.txt

4. To create a Citrix NetScaler certkey, which is the in-memory object that holds the server certificate information for SSL handshakes and bulk encryption, at the NetScaler command line, type:
   add ssl certkey test-certkey -cert /nsconfig/ssl/test-server.cer -key /nsconfig/ssl/test-server.key

5. To bind the certkey object to the SSL virtual server, at the NetScaler command line, type:
   bind ssl vserver <vServerName> -certkeyName <cert_name>

I have received a Citrix NetScaler appliance on which Citrix NetScaler software release 9.0 is installed. I have noticed an additional license file on the appliance. Is there any change in the licensing policy starting with Citrix NetScaler software release 9.0?

Yes. Starting with Citrix NetScaler software release 9.0, the appliance might not have a single license file. The number of license files depends on the Citrix NetScaler software release edition. For example, if you have installed the Enterprise edition, you might need additional license files for the full functionality of the various features. However, if you have installed the Platinum edition, the appliance has only one license file.

How do I export the certificate from Internet Information Service (IIS)?

There are many ways to do this, but by using the following method the appropriate certificate and private key for the Web site are exported. This procedure **must** be performed on the actual IIS server.

1. Open the Internet Information Services (IIS) Manager administration tool.
2. Expand the Web Sites node and locate the SSL-enabled Web site that you want to serve through the Citrix NetScaler.
3. Right-click this Web site and click Properties.
4. Click the Directory Security tab and, in the Secure Communications section of the window, select the View Certificate box.
5. Click the Details tab, and then click Copy to File.
6. On the Welcome to the Certificate Export Wizard page, click Next.
7. Select Yes, export the private key and click Next.

Note: The private key MUST be exported for SSL Offload to work on the Citrix NetScaler

8. Make sure that the Personal Information Exchange -PKCS #12 radio button is selected, and select *only* the Include all certificates in the certification path if possible check box. Click Next.
9. Enter a password and click Next.
10. Enter a file name and location, and then click Next. Give the file an extension of .PFX.
11. Click Finish.

How do I convert the PKCS#12 certificate and install it on the Citrix NetScaler?

1. Move the exported .PFX certificate file to a location from where it may be copied to the Citrix NetScaler (that is, to a machine that permits SSH access to the management interface of a Citrix NetScaler appliance). Copy the certificate to the appliance by using a secure copy utility such as SCP.
2. Access the BSD shell and convert the certificate (for example, cert.PFX) to .PEM format:
   root@ns# openssl pkcs12 -in cert.PFX -out cert.PEM
3. To make sure that the converted certificate is in correct x509 format, verify that the following command produces no error:
   root@ns# openssl x509 -in cert.PEM -text
4. Verify that the certificate file contains a private key. Begin by issuing the following command:
   root@ns# cat cert.PEM

   Verify that the output file includes an RSA PRIVATE KEY section.

   -----BEGIN RSA PRIVATE KEY-----
   Mkm^s9KMs9023pz/s...
   -----END RSA PRIVATE KEY-----
   The following is another example of an RSA PRIVATE KEY section:

   Bag Attributes
   1.3.6.1.4.1.311.17.2: <No Values>
   localKeyID: 01 00 00 00
   Microsoft CSP Name: Microsoft RSA SChannel Cryptographic
   Provider
   friendlyName:
   4b9cef4cc8c9b849ff5c662fd3e0ef7e_76267e3e-6183-4d45-886e-6e067297b38f

   Key Attributes
   X509v3 Key Usage: 10
   -----BEGIN RSA PRIVATE KEY-----
   Proc-Type: 4,ENCRYPTED
   DEK-Info: DES-EDE3-CBC,43E7ACA5F4423968
   pZJ2SfsSVqMbRRf6ug37Clua5gY0Wld4frPIxFXyJquUHr31dilW5ta3hbIaQ+Rg

   ... (more random characters)
   v8dMugeRplkaH2Uwt/mWBk4t71Yv7GeHmcmjafK8H8iW80ooPO3D/ENV8X4U/tIh

   5eU6ky3WYZ1BTy6thxxLIwAullynVXZEfINLxq1oX+ZYl6djgjE3qg==
   -----END RSA PRIVATE KEY-----
   The following is a SERVER CERTIFICATE section:

   Bag Attributes

localKeyID: 01 00 00 00
friendlyName: AG Certificate
subject=/C=AU/ST=NSW/L=Wanniassa/O=Dave Mother
Asiapacific/OU=Support/CN=davemother.food.lan
issuer=/DC=lan/DC=food/CN=hotdog
-----BEGIN CERTIFICATE-----
MIIFiTCCBHGgAwIBAgIKCGryDgAAAAAHzANBgkqhkiG9w0BAQUFADA8MRMwEQYK

... (more random characters) 5pLDWYVHhLkA1pSxvFjNJHRSIydWHc5ltGyKqIUcBezVaXyel94pNSUYx07NpPV/

MY2ovQyQZM8gGe3+lGFum0VHbv/y/gB9HhFesog=
-----END CERTIFICATE-----
The following is an INTERMEDIATE CA CERTIFICATE section:

Bag Attributes: <Empty Attributes>
subject=/DC=lan/DC=food/CN=hotdog
issuer=/DC=lan/DC=food/CN=hotdog
-----BEGIN CERTIFICATE-----
MIIESDCCAzCgAwIBAgIQah20fCRYTY9LRXYMIRaKGjANBgkqhkiG9w0BAQUFADA8

... (more random characters) Nt0nksawDnbKo86rQcNnY5xUs7c7pj2zxj/IOsgNHUp5W6dDI9pQoqFFaDk=

-----END CERTIFICATE-----
Further Intermediate CA certificates may follow, depending on the certification path of the exported certificate.

5. Open the .PEM file in a text editor
6. Locate the first line of the .PEM file and the first instance of the following line, and copy those two lines and all the lines between them:
   -----END CERTIFICATE-----

   Note: Make sure that last copied line is the first -----END CERTIFICATE----- line in the .PEM file.
7. Paste the copied lines into a new file. Call the new file something intuitive, such as cert-key.pem. This is the certificate-key pair for the server hosting the HTTPS service. This file should contain both the section labeled RSA PRIVATE KEY and the section labeled SERVER CERTIFICATE in the example above.
   Note: The certificate-key pair file contains the private key and must therefore be kept secure.
8. Locate any subsequent sections beginning with -----BEGIN CERTIFICATE----- and ending with ---END CERTIFICATE----, and copy each such section to a separate new file.
   These sections correspond to certificates of trusted CAs that have been included in the certification path. These sections should be copied and pasted into new individual files for these certificates. For example, the INTERMEDIATE CA CERTIFICATE section of the example above should be copied and pasted into a new file).

   For multiple intermediate CA certificates in the original file, create new files for each intermediate CA certificate in the order in which they appear in the file. Keep track (using appropriate filenames) of the order in which the certificates appear, as they need to be linked together in the correct order in a later step.

9. Copy the certificate-key file (cert-key.pem) and any additional CA certificate files into the /nsconfig/ssl directory on the Citrix NetScaler.
10. Exit the BSD shell and access the Citrix NetScaler prompt.
11. Follow the steps in "Install the certificate-key files on the appliance" to install the key/certificate once uploaded on the

device.

How do I convert the PKCS#7 certificate and install it on the NetScaler appliance?

You can use OpenSSL to convert a PKCS #7 Certificate to a format recognizable by the NetScaler appliance. The procedure is identical to the procedure for PKCS #12 certificates, except that you invoke OpenSSL with different parameters. The steps for converting PKCS #7 certificates are as follows:

1. Copy the certificate to the appliance by using a secure copy utility, such as SCP.
2. Convert the certificate (for example, cert.P7B) to PEM format:

   ```
   > openssl pkcs7 -inform DER -in cert.p7b -print_certs -text -out cert.pem
   ```

3. Follow steps 3 through 7 as described in the answer to Q32 for PKCS #12 certificates.

Note: Before loading the converted PKCS #7 certificate to the appliance, be sure to verify that it contains a private key, exactly as described in step 3 for the PKCS #12 procedure. PKCS #7 certificates, particularly those exported from IIS, do not typically contain a private key.

When I bind a cipher to a virtual server or service by using the bind cipher command, I see the error message "Command deprecated."

The command for binding a cipher to a virtual server or service has changed.

Use the bind ssl vserver <vsername> -ciphername <ciphername> command to bind an SSL cipher to an SSL virtual server.

Use the bind ssl service <serviceName> -ciphername <ciphername> command to bind an SSL cipher to an SSL service.

Note: New ciphers and cipher groups are added to the existing list and not replaced.

Why can't I create a new cipher group and bind ciphers to it by using the add cipher command?

The add cipher command functionality has changed in release 10. The command only creates a cipher group. To add ciphers to the group, use the bind cipher command.

How do I install the OpenSSL toolkit?

See the article at http://support.citrix.com/article/ctx106627.

How do I use OpenSSL to convert certificates between PEM and DER?

To use OpenSSL, you must have a working installation of the OpenSSL software and be able to execute Openssl from the command line.
x509 certificates and RSA keys can be stored in a number of different formats.

Two common formats are DER (a binary format used primarily by Java and Macintosh platforms) and PEM (a base64 representation of DER with header and footer information, which is used primarily by UNIX and Linux platforms). There is also an obsolete NET (Netscape server) format that was used by earlier versions of IIS (up to and including 4.0) and various other less common formats that are not covered in this article.

A key and the corresponding certificate, as well as the root and any intermediate certificates, can also be stored in a single PKCS#12 (.P12, .PFX) file.

Procedure

Use the Openssl command to convert between formats as follows:

1. To convert a certificate from PEM to DER:

x509 -in input.crt -inform PEM -out output.crt -outform DER

2. To convert a certificate from DER to PEM:

   x509 -in input.crt -inform DER -out output.crt -outform PEM

3. To convert a key from PEM to DER:

   rsa -in input.key -inform PEM -out output.key -outform DER

4. To convert a key from DER to PEM:

   rsa -in input.key -inform DER -out output.key -outform PEM

Note: If the key you are importing is encrypted with a supported symmetric cipher, you are prompted to enter the pass-phrase.
Note: To convert a key to or from the obsolete NET (Netscape server) format, substitute NET for PEM or DER as appropriate. The stored key is encrypted in a weak unsalted RC4 symmetric cipher, so a pass-phrase will be requested. A blank pass-phrase is acceptable.

What are the important numbers to remember?

1. Create Certificate Request:
   - Request File Name: Maximum 63 characters
   - Key File Name: Maximum 63 characters
   - PEM Passphrase (For Encrypted Key): Maximum 31 characters
   - Common Name: Maximum 63 characters
   - City: Maximum 127 characters
   - Organization Name: Maximum 63 characters
   - State/Province Name: Maximum 63 characters
   - Email Address: Maximum 39 Characters
   - Organization Unit: Maximum 63 characters
   - Challenge Password: Maximum 20 characters
   - Company Name: Maximum 127 characters
2. Create Certificate:
   - Certificate File Name: Maximum 63 characters
   - Certificate Request File Name: Maximum 63 characters
   - Key File Name: Maximum 63 characters
   - PEM Passphrase: Maximum 31 characters
   - Validity Period: Maximum 3650 days
   - CA Certificate File Name: Maximum 63 characters
   - CA Key File Name: Maximum 63 characters
   - PEM Passphrase: Maximum 31 characters
   - CA Serial Number File: Maximum 63 characters
3. Create and Install a Server Test Certificate:
   - Certificate File Name: Maximum 31 characters
   - Fully Qualified Domain Name: Maximum 63 characters
4. Create Diffie-Hellman (DH) key:
   - DH Filename (with path): Maximum 63 characters
   - DH Parameter Size: Maximum 2048 bits
5. Import PKCS12 key:

- Output File Name: Maximum 63 characters
- PKCS12 File Name: Maximum 63 characters
- Import Password: Maximum 31 characters
- PEM Passphrase: Maximum 31 characters
- Verify PEM Passphrase: Maximum 31 characters

6. Export PKCS12
   - PKCS12 File Name: Maximum 63 characters
   - Certificate File Name: Maximum 63 characters
   - Key File Name: Maximum 63 characters
   - Export Password: Maximum 31 characters
   - PEM Passphrase: Maximum 31 characters
7. CRL Management:
   - CA Certificate File Name: Maximum 63 characters
   - CA Key File Name: Maximum 63 characters
   - CA Key File Password: Maximum 31 characters
   - Index File Name: Maximum 63 characters
   - Certificate File Name: Maximum 63 characters
8. Create RSA Key:
   - Key Filename: Maximum 63 characters
   - Key Size: Maximum 4096 bits
   - PEM Passphrase: Maximum 31 characters
   - Verify Passphrase: Maximum 31 characters
9. Create DSA Key:
   - Key Filename: Maximum 63 characters
   - Key Size: Maximum 4096 bits
   - PEM Passphrase: Maximum 31 characters
   - Verify Passphrase: Maximum 31 characters
10. Change advanced SSL settings:
    - Maximum CRL memory size: Maximum 1024 Mbytes
    - Encryption trigger timeout (10 mS ticks): Maximum 200
    - Encryption trigger packet count: Maximum 50
    - OCSP cache size: Maximum 512 Mbytes
11. Install Certificate:
    - Certificate-Key pair Name: Maximum 31 characters
    - Certificate File Name: Maximum 63 characters
    - Private Key File Name: Maximum 63 characters
    - Password: Maximum 31 characters
    - Notification Period: Maximum 100
12. Create Cipher Group:
    - Cipher Group Name: Maximum 39 characters
13. Create CRL:
    - CRL Name: Maximum 31 characters
    - CRL File: Maximum 63 characters
    - URL: Maximum 127 characters
    - Base DN: Maximum 127 characters
    - Bind DN: Maximum 127 characters
    - Password: Maximum 31 characters
    - Day(s): Maximum 31

14. Create SSL Policy:
    - Name: Maximum 127 characters
15. Create SSL Action:
    - Name: Maximum 127 characters
16. Create OCSP Responder:
    - Name: Maximum 32 characters
    - URL: Maximum 128 characters
    - Batching Depth: Maximum 8
    - Batching Delay: Maximum 10000
    - Produced At Time Skew: Maximum 86400
    - Request Time-out: Maximum120000
17. Create Virtual Server:
    - Name: Maximum 127 characters
    - Redirect URL: Maximum 127 characters
    - Client Time-out: Maximum 31536000 secs
18. Create Service:
    - Name: Maximum 127 characters
    - Idle Time-out (secs):
      Client: Maximum 31536000

      Server: Maximum 31536000
19. Create Service Group:
    - Service Group Name: Maximum 127 characters
    - Server ID: Maximum 4294967295
    - Idle Time-out (secs):
      Client: Maximum value 31536000

      Server: Maximum 31536000
20. Create Monitor:
    - Name: Maximum 31 characters
21. Create Server:
    - Server Name: Maximum 127 characters
    - Domain Name: Maximum 255 characters
    - Resolve Retry: Maximum 20939 secs

Installing, Upgrading, and Downgrading

Aug 13, 2013

What is the use of the zebos.conf file available in a NetScaler release?

A NetScaler appliance uses Zebos as the routing suite. The zebos.conf file available in a NetScaler release is the configuration file for Zebos.

I want to change the SSH port (22) on the NetScaler appliance to some other port. Is it possible to change the SSH port on the appliance?

Yes. You can change the SSH port on the NetScaler appliance by editing the sshd_config file in the /nsconfig directory. If the file does not exist in the /nsconfig directory, copy it from the /etc directory.
In the sshd_config file, edit the entry for Port 22 to Port <Number>, where <Number> is the target port number. If you do not want to restart the appliance and make the changes effective, terminate the sshd process by using the kill command, and then restart the process.

The flash directory is missing from the NetScaler appliance. What procedure should I follow to mount the flash directory?

To mount the flash directory, do the following:

1. Start the NetScaler appliance in single-user mode.
   When the appliance starts, the following message appears:

   Hit [Enter] to boot immediately, or any other key for command prompt. Booting [kernel] in 10 seconds...”
   Hit space and you should see the following prompt:

   Type '?' for a list of commands, 'help' for more detailed help.

2. Enter the following command to start FreeBSD in single-user mode:
   boot –s

   After the appliance starts, the following message appears:

   Enter full pathname of shell or RETURN for /bin/sh:

3. Press Enter to display the # prompt.
4. Run the following command to mount the flash directory:
   mount /dev/ad0s1a /flash

   Note: If the preceding command displays an error message about permissions, run the following command to check the disk for consistency:
   fsck /dev/ad0s1a

   Run the mount command again to mount the flash directory.

5. Restart the appliance.
6. From the shell prompt, run the following command to verify that the flash directory is mounted:
   df -kh

I want to log on to the NetScaler appliance without entering the password. Is it possible to configure SSH on the appliance to allow that?

Yes. You can configure SSH on the NetScaler appliance to log on without a password. However, you must provide your user name. To configure SSH for logging in without a password, do the following:

1. Run the following command to generate the public and private keys:
   ```
   # ssh-keygen -t rsa
   ```
2. Run the following command to copy the id_rsa.pub file to the .ssh directory of the remote host that you want to log on to:
   ```
   # scp id_dsa.pub <user>@<remote_host>/.ssh/id_dsa.pub
   ```
3. Log on to the remote host.
4. Change to the .ssh directory.
5. Run the following commands to add the public key of the client to the known public keys:
   ```
   # cat id_dsa.pub >> authorized_keys2

   # chmod 640 authorized_keys2

   # rm id_dsa.pub
   ```

What is the procedure to reset the NetScaler appliance BIOS? Under what circumstances should I reset the BIOS?

To reset BIOS of the NetScaler appliance, complete the following procedure:

1. Connect to the appliance through the serial port.
2. Start the appliance and press Delete as soon as the boot-up process starts.
   Pressing Delete during the POST process displays the appliance's BIOS settings.
3. Activate the Exit page of the BIOS settings.
4. Select the Load Optimal Defaults option.
   The Load Optimal Settings message box appears.
5. Select OK.
6. Make the following changes to the BIOS settings on the various tabs:

Tab	Group	Component	Set to...
Advanced	SuperIO Configuration	Parallel Port Address	Disabled
	Floppy Configuration	Floppy A	Disabled
	Boot Settings Configuration	Quiet Boot	Disabled
		PS/2 Mouse Support	Disabled
		Parity Check	Enable
	Remote Access Configuration	Remote Access	COM1
		Serial Port Mode	9600 8'n'1

Tab	Group	Component	Set to...
Chipset		Hyper-threading	Disabled
PCI PnP		Allocate IRQ to PCIVGA	NO
		USB Function	Disabled
		Legacy USB Support	Disabled
Power		ACPI Aware OS	NO
		Power Management	Enabled

Note: The BIOS options differs by appliance model.

7. Activate the Exit page of the BIOS settings.
8. Select Save changes and Exit.
9. Select OK to confirm.
10. Verify that the appliance starts cleanly and the serial console displays output after the appliance starts.

You need to reset BIOS when the serial console does not respond. This usually happens after you upgrade the appliance and the serial console is disabled. However, you can still access the appliance by using the telnet or SSH utility.

I need to reset the NetScaler appliance to the factory defaults. What procedure should I follow?

To reset the NetScaler appliance to the factory defaults, you need to reset two environments: the NetScaler application environment and the base FreeBSD environment.

To reset the NetScaler application environment of the appliance to the factory defaults, do the following:

1. Make a backup of the appliance's /nsconfig/ns.conf.
2. Delete the /nsconfig/ns.conf file.
3. Restart the appliance.

To reset the FreeBSD environment of the appliance to the factory defaults, do the following:

1. Install a fresh NetScaler code image on the appliance. This overwrites a number of FreeBSD-level configuration files with default values.
2. Delete any users and groups that are added to the appliance, that is, all except the default users.
3. Delete the **/etc/resolv.conf** file.
4. Delete the entries that you have added to the **/etc/hosts** file.
5. If the **/etc/rc.netscaler** file exists, delete it.
6. Open the **/etc/nsperm_group_suser** file and make sure that all IOCTL entries are comment entries.
7. Open the **/etc/rc.conf** file and make sure that the **syslogd_enable=NO** entry is not changed to **syslogd_enable=YES**.
8. Open the **/etc/syslog.conf** file and make sure that there are no additional entries in the file.
9. Delete the contents of the **/var/nslog**, **/var/nstrace**, and **/var/crash** files.
10. If the **syslog** process is enabled on the appliance and the appliance creates log files locally, delete the contents of the log files listed in the **/etc/syslog.conf** file. The files are created in the /var/log directory. For example, if **syslog** process writes system events to the **/var/log/events** file, and sslvpn access events to the **/var/log/sslvpnevents** file, delete

these files.

The appliance displays a message similar to the "Jun 21 12:20:18 ns /flash/ns-10.0-47.15: [1/2]dc0: NIC hang condition #663: TX 10000/10000, RX 0, HF 0" message on the console. What is the meaning of this message?

The message consists of the following components (shown here as examples):

- #663: Number of times this condition has occurred on the appliance.
- TX 10000/10000: Number of packets that the appliance attempted to transmit, and number of packets transmitted. If both numbers are the same, as in this example, the NIC transmitted all the packets that the appliance attempted to transmit.
- RX 0: Number of packets received. In this example, no packet was received.
- HF0: Number of hardware issues reported by the NIC. In this example, the NIC did not report any hardware issue.

If the appliance does not receive any packets, it reports a hang condition, because on a network it is very unlikely not to receive any packets. However, if the appliance is plugged into quite interface, you can ignore this error message.

After I upgraded the NetScaler release on the appliance, the appliance still displays the earlier release/build. What could be the reason?

The appliance displays the software version number from the /flash/boot/loader.conf file. If the kernel entry for the current NetScaler release is missing from that file, the appliance displays the last NetScaler release version for which the entry was available.

To resolve this issue, do the following:

1. Verify that the kernel file exists in the /nsconfig directory.
2. Check the /flash/boot/loader.conf file for an entry for the kernel.
   (You can expect the entry for the kernel of the release/build that you just installed to be missing from the file.)
3. Open the loader.conf file in a text editor, such as the vi editor, and update the kernel entry for the new release/build.
4. Save and close the file.
5. Repeat step 2 through step 4 for the /flash/boot/loader.conf.local file.
6. Update the release/build entry in the ns.conf file.
7. Restart the appliance.

Since I upgraded the NetScaler release on the appliance, the LCD display on the front panel of the appliance displays the out of service message or does not display anything. How can I resolve this issue?

Run the following command from the appliance's shell prompt:
/netscaler/nslcd –k

I have upgraded the NetScaler release/build. However, after the upgrade process, the appliance fails to start. Can I downgrade the appliance's software to the previous release/build?

Yes. You can start the appliance with the kernel.old kernel file. When you restart the appliance, press the F1 key as soon as the appliance console displays the Press F1 message. Type kernel.old and press Enter.

After upgrading the NetScaler release on the appliance, I accidently deleted the kernel file from the /flash directory. As a result, I am not able to start the appliance. Is there a method for starting the appliance in this situation?

Yes. You can start the appliance by using the kernel.GENERIC kernel file, as follows:

1. When you restart the appliance, press the F1 key as soon as the appliance console displays the Press F1 message.
2. Type kernel.GENERIC and press Enter.
3. Login as the root user.

4. Reinstall the NetScaler release.
5. Restart the appliance.

I have received a NetScaler appliance with the latest NetScaler release installed on it. However, I want to downgrade the software release. Can I do so?

No. If you attempt to downgrade the software release, the appliance might not work as expected, because the ns.conf file of the later release might not be compatible with the earlier release, and the appliance might restore to the factory settings.

When downgrading the NetScaler release, I followed the instructions. However, the appliance displays the following message:

```
root@LBCOL03B# ./installns
installns version (10.0-47.7) kernel (ns-10.0-47.7.gz)
Note:
Installation may pause for up to 3 minutes while data is written to the flash.
Caution:
Do not interrupt the installation process.
Doing so may cause the system to become unusable.
Installation will proceed in 5 seconds, CTRL-C to abort
No Valid Netscaler Version Detected
root@LBCOL03B#
```

Am I doing something incorrectly?

This issue could be the result of incorrect version information in the ns.conf file. To resolve this issue, open the ns.conf file in a text editor, such as the vi editor. Update the release-specific entry in the ns.conf file to #NS<Release_No> Build <Build_No>. Here, <Release_No> is the NetScaler release number to which you want to downgrade the software, and the <Build_No> is the build number of the software release to which you want to downgrade the software.

After upgrading the appliance software to NetScaler release 10.0, I am not able to log on to the appliance, and the following message is appears:

```
login: nsroot
Password:
connect: No such file or directory
nsnet_connect: No such file or directory
Login incorrect
```

I tried to resolve this issue by using the password recovery procedure, but I was not successful. Have I done something incorrectly?

You cannot resolve this issue by using the password recovery procedure. NetScaler releases 8.0 and later use the new licensing system, based on the Imgrd daemon, which runs during the startup procedure. For this daemon to work properly, the host name of NetScaler appliance, which is set in the /nsconfig/rc.conf file, must be resolved by a name server to the NetScaler IP address. Alternately, you can create a hosts file in the /nsconfig directory and add the 127.0.0.1 <Host_Name> entry in file.

Additionally, make sure that you have copied the license files to the /nsconfig/license/ directory.

During an upgrade of a high availability pair, the following message appears repeatedly:

```
<auth.err> ns sshd[5035]: error: Invalid username or password
```


What could be the reason?

This error message appears when the appliances involved in the high availability pairing have either a different NetScaler release or a different builds of the same release installed. The appliances can have different version installed if you have upgraded or downgraded one appliance but not the other.

I want to change the netmask of the NetScaler IP address on a NetScaler appliance. Can I do so without causing an outage?

Changing the netmask of the NetScaler IP might result in a short outage. Make sure that you change the netmask on the secondary appliance, and then break the high availability pairing. Check the functionality of the appliance. If everything works as expected, rebuild the high availability pairing.

To change the netmask on the appliance, run the configns command from the CLI prompt, and then choose the second option in the menu.

I have configured a High Availability pair of NetScaler appliances. After upgrading the software release from a beta release to a final release, I noticed that some of the appliance configurations are missing. Can I retrieve the lost configurations?

You can use the following procedure to restore the configuration:

1. Log on to the command line interface of the primary appliance.
2. Run the following commands:

   save config

   shell

   #cp /nsconfig/ns.conf /nsconfig/ns.conf.bkup

   The ns.conf.bkup file is a backup for the running configuration.

3. Upgrade software of both the appliances to the final release.
4. Log on to the command line interface of the primary appliance.

Surge Protection

Jan 13, 2014

On the NetScaler appliance, I have configured the Surge Queue feature with a MaxClients value set on a service. After browsing a web page, is the user connection placed in the surge queue when a user opens the next page?

No. The connection is not placed in the surge queue, because the user accesses the same server. The persistence method configured for the virtual server ensures that the connection is maintained with the same server until the connection times out, at which point and server resource becomes available.

When a web browser sends multiple requests, is it possible for a request to be placed in surge queue?

No. Until the connection times out, making the server resource available for new connections, requests are not placed in the surge queue. If you make another request after the connection times out, a new connection is established and the request is put into the surge queue if the server resource is not available.

Can I specify a limit for the surge queue on the appliance?

No. You cannot limit the surge queue. The size of the surge queue depends on the available memory of the appliance.

What happens when the appliance has no memory available for the surge queue?

When the appliance approaches the memory limit, it triggers the memory recovery process. If still the appliance does not have enough memory, it stops accepting client connections.

What is the purpose of the MaxClient value?

The MaxClient value guards the server against network attacks by allowing only the specified number of client connections to the server. If this value is not set, the appliance passes all client connections to the server. As a result, the server might be slow in responding to a client request or might not respond at all.

NetScaler Solutions

Aug 23, 2013

NetScaler solutions simplify the task of setting up frequently deployed configurations. Check this space from time to time for additional solutions.

This document includes the following solutions:

- Setting Up NetScaler for XenApp/XenDesktop
- Integrating NetScaler ADCs with Cisco ACI
- RISE Integration: NetScaler ADC and Cisco Nexus 7000 Series Switch
- Web Interface

Setting Up NetScaler for XenApp/XenDesktop

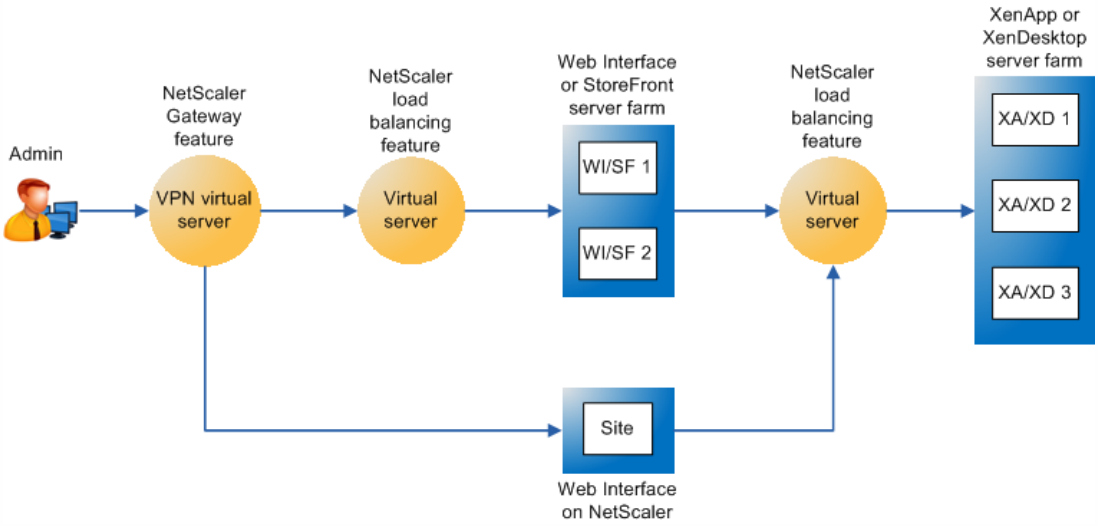
Dec 10, 2013

A NetScaler appliance can provide load balanced, secure remote access to your XenApp/XenDesktop applications. You can use the NetScaler load balancing feature to distribute traffic across the XenApp/XenDesktop servers, and the NetScaler Gateway feature to provide secure remote access to the servers. NetScaler can also accelerate and optimize the traffic flow and offer visibility features that are useful for XenApp/XenDesktop deployments.

The configurations that are required to be performed on the NetScaler are consolidated in a wizard that simplifies the deployment. You can also apply the following preset configurations:

- Optimization settings such as TCP profiles, compression, caching, and SSL quantum settings.
- Security settings such as application firewall profiles and policies.
- Visibility settings such as HDX Insight policies.

Figure 1. NetScaler Appliance in XenApp/XenDesktop Setup



The above figure shows the components involved in this deployment:

- **NetScaler Gateway.** Provides the URL for user access, and provides security by authenticating the users.
- **NetScaler load balancing virtual server.** Load balances the traffic for the Web Interface or StoreFront servers. You can also deploy a load balancing virtual server in front of the XenApp/XenDesktop servers to load balance key components such as XML Broker and Desktop Delivery Controller (DDC) server.
- **Web Interface or StoreFront or Web Interface on NetScaler.** Provides the interface through which you can access the applications.

  Note: Web Interface on NetScaler (WIonNS) is a customization of the Web Interface product, hosted on the NetScaler appliance.
- **XenApp/XenDesktop.** Provides the applications that your users want to access.

Prerequisites

Before using the wizard, make sure of the following:

- XenApp/XenDesktop servers are configured and available.
- Web Interface, StoreFront, or Web Interface on NetScaler servers are configured and available.
- You have a working knowledge of NetScaler Gateway, NetScaler, XenApp, XenDesktop, and StoreFront/Web Interface/Web Interface

on NetScaler. For more information, see "Citrix eDocs."

To set up the NetScaler for XenApp/XenDesktop by using the wizard

1. Log on to the NetScaler appliance and, on the Configuration tab, navigate to Traffic Management > Load Balancing.
2. In the details pane, under XenApp/XenDesktop, click Set Up NetScaler for XenApp/XenDesktop.
   Note: If the setup exists on the NetScaler, click the Edit link corresponding to each of the section that you want to modify.
3. Select the XenApp/XenDesktop deployment type.
   Note: The wizard supports only the single-hop deployment of XenApp/XenDesktop.
4. Select the product (StoreFront, Web Interface, or Web Interface on NetScaler) that in your deployment provides the interface for access to the XenApp/XenDesktop applications.
5. Set up secure remote access.
   1. In the NetScaler Gateway Settings section, specify the details for the VPN virtual server.
   2. In the Certificate section, choose an existing certificate or install a new certificate.
   3. In the Authentication Settings section, configure the primary authentication mechanism to be used and specify the server details. You can also configure secondary authentication to provide two-factor authentication.
      Note: While configuring the primary authentication mechanism, you can select the Load Balancing check box to distribute traffic among authentication servers. In the address field that appears, specify the IP address to assign to the load balancing virtual server.
6. Set up the interface used to access the applications. In the Web Interface, StoreFront, or Web Interface on NetScaler section, do the following:
   1. Specify the details of the server that provides the interface for accessing the applications.
   2. Select the Load Balancing check box to distribute load among the servers. In the address field that appears, specify the IP address to assign to the load balancing virtual server.
      Note: If Web Interface on NetScaler is selected in this wizard, but it is not installed on the NetScaler appliance, you are prompted to upload the TAR and JRE files. For more information, see "Installing the Web Interface."
7. Specify the XenApp/XenDesktop server(s) from which the applications are to be accessed. In the Xen Farm section, do the following:
   1. Provide details of the servers from which your users want to access applications.
   2. Select the Load Balancing check box to distribute load among the servers. In the address field that appears, specify the IP address to assign to the load balancing virtual server.
8. Configure optimization, security, and visibility on the NetScaler appliance.
   - In the Optimization section, click Apply. The following configurations are executed internally:

     TCP Profile

     > set vpn vserver ag_vsvr1 -tcpProfileName nstcp_default_XA_XD_profile
     > set servicegroup WI_servicegroup -tcpProfileName nstcp_default_XA_XD_profile
     > set servicegroup SF_servicegroup -tcpProfileName nstcp_default_XA_XD_profile
     > set servicegroup XA_Primary_Broker_servicegroup -tcpProfileName nstcp_default_XA_XD_profile
     > set servicegroup XA_Secondary_Broker_servicegroup -tcpProfileName nstcp_default_XA_XD_profile
     > set servicegroup XD_servicegroup -tcpProfileName nstcp_default_XA_XD_profile

     Compression

     > enable ns feature cmp
     > set servicegroup WI_servicegroup -cmp on
     > set servicegroup SF_servicegroup -cmp on
     > set servicegroup XA_Primary_Broker_servicegroup -cmp on
     > set servicegroup XA_Secondary_Broker_servicegroup -cmp on
     > set servicegroup XD_servicegroup -cmp on

     Caching

     > enable ns feature IC
     > add cache contentgroup cache_group_XA-XD
     > set cache parameter -memLimit 100
     > add cache policy cache_pol1 -rule TRUE -action CACHE -storeInGroup cache_group_XA-XD

> bind cache global XA_XD_10.102.87.108_cachepol -priority 10 -gotoPriorityExpression END -type REQ_DEFAULT

SSL quantum settings

> set ssl parameter -quantumSize 4 -sslTriggerTimeout 10 -encryptTriggerPktCount 10 -pushEncTriggerTimeout 10

- In the Security section, click Apply.

  Note: The security settings are not applicable for this release.
- In the Visibility section, click Apply. The following configurations are executed internally:

  > enable feature Appflow
  > set vpn vserver ag_vsvr1 -appflowLog ENABLED

  Note: Make sure that the appliance is added to the NetScaler Insight Center appliance.

9. Click Done to complete the configuration.

Integrating NetScaler ADCs with Cisco ACI

Apr 04, 2016

As businesses quickly move to make the datacenter more agile, the application centric automation and virtualization of both hardware and software infrastructure become increasingly important. Cisco Application Centric Infrastructure (ACI) supplies the critical link between business-based requirements for applications and the infrastructure that supports them. The Citrix NetScaler ADC connects infrastructure and applications and makes their configuration available to the Cisco Application Policy Infrastructure (APIC) through integration.

Citrix NetScaler and Cisco ACI enable datacenter and cloud administrators to holistically control L2-L7 network services in a unified manner, through seamless insertion and automation of best-in-class NetScaler services into next-generation datacenters built on Cisco's ACI Architectures. NetScaler leverages the Cisco Application Policy Infrastructure Controller (APIC) to programmatically automate network provisioning and control on the basis of application requirements and policies for both datacenter and enterprise environments.

Click here for more information on how to deploy NetScaler ADCs in Cisco ACI.

/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

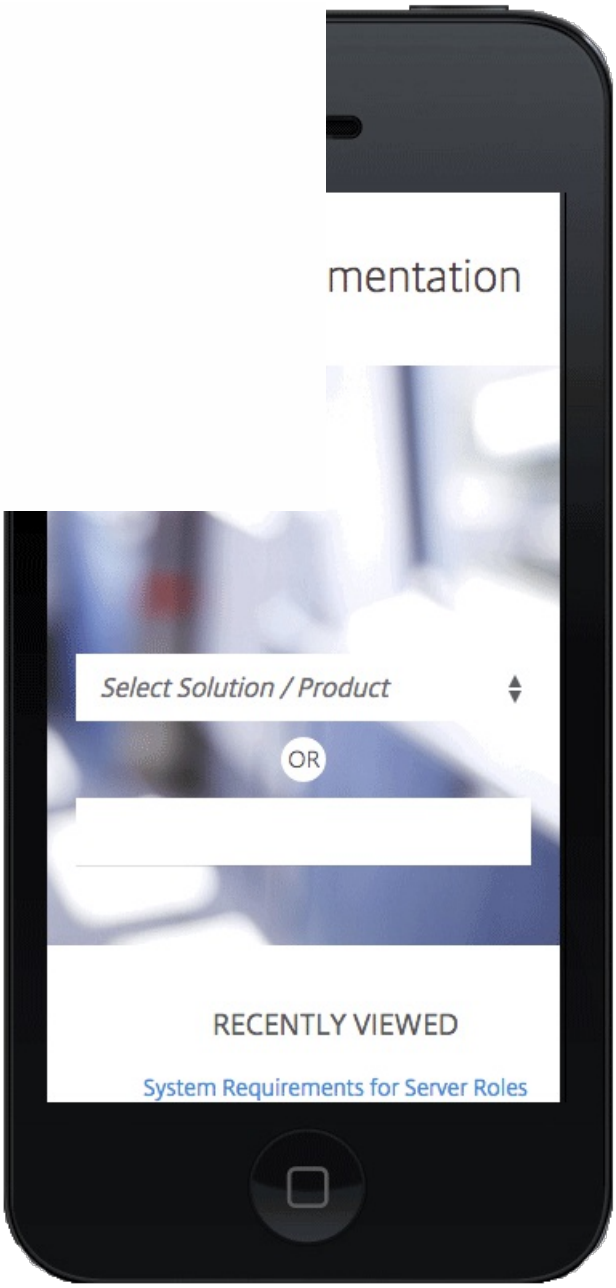
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

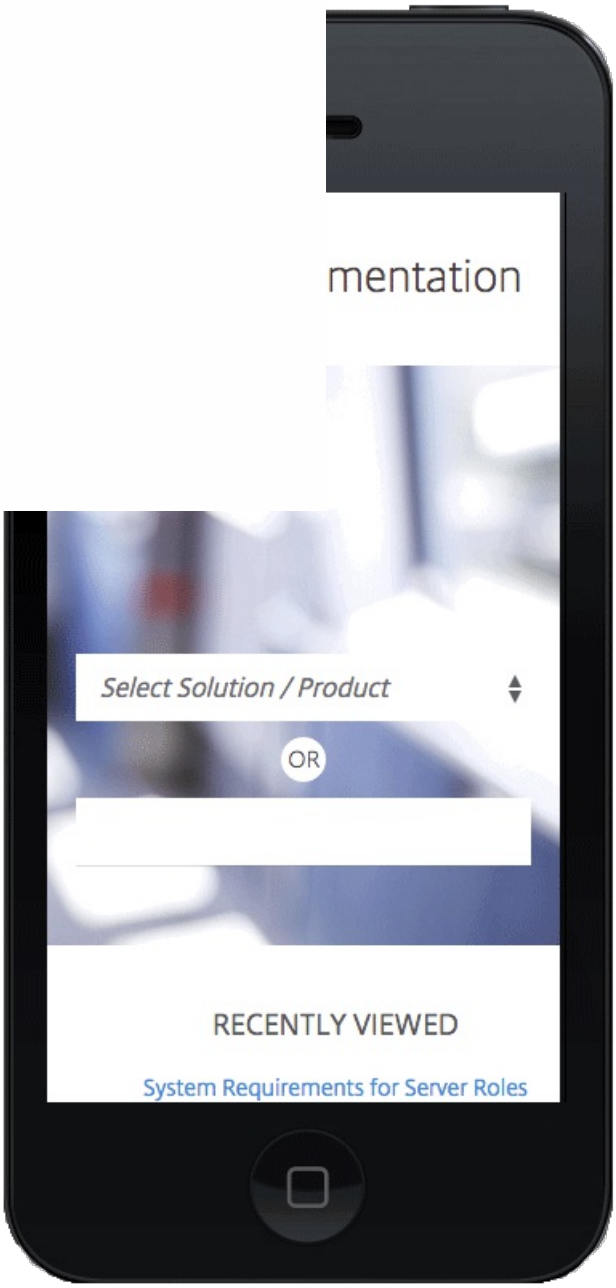
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

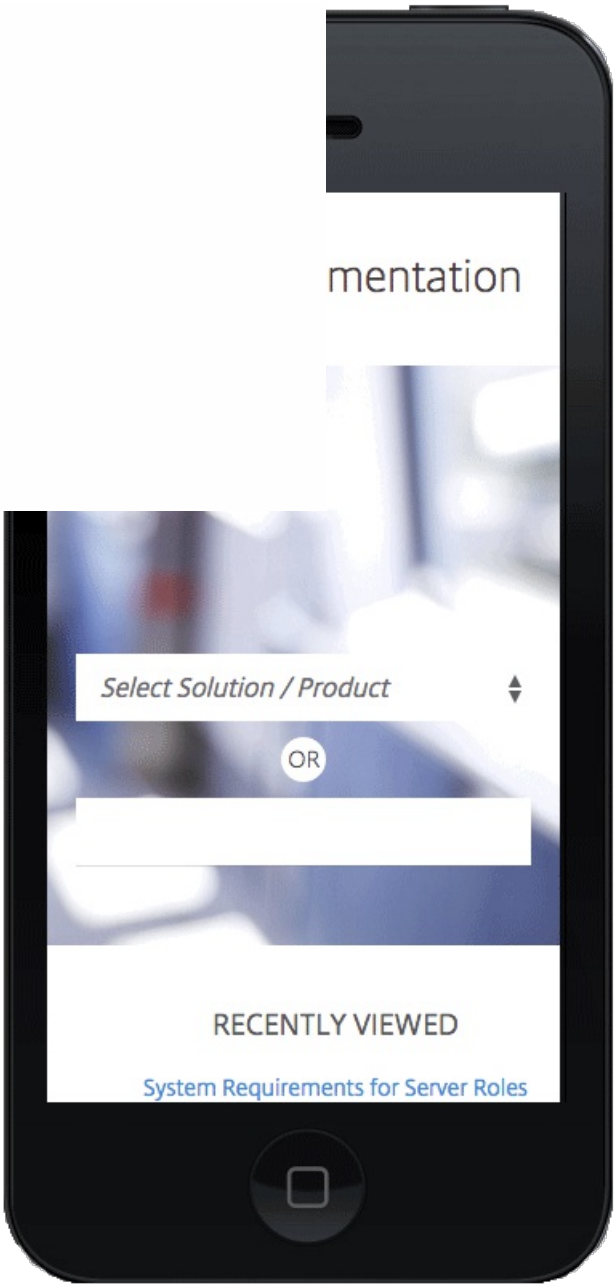
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

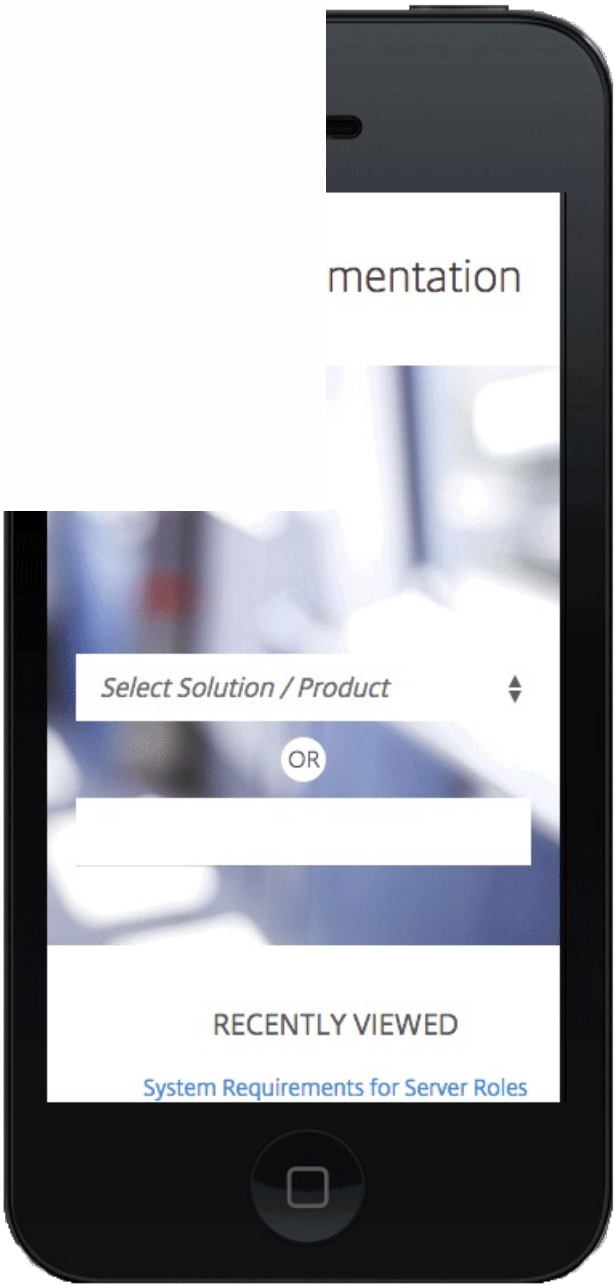
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

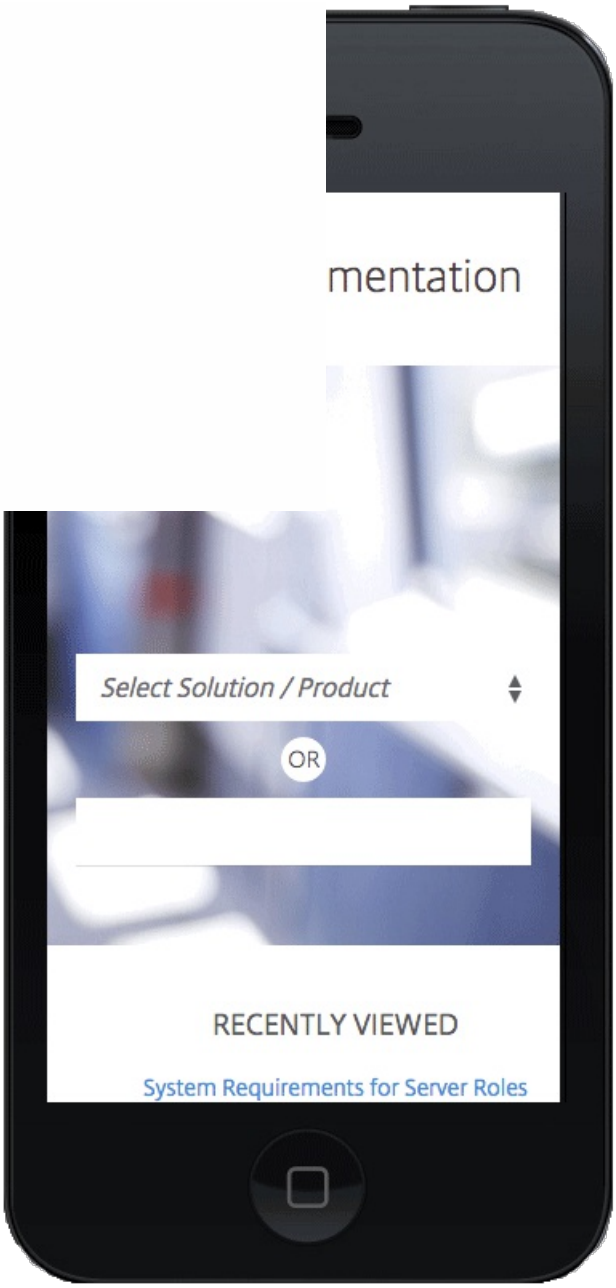
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

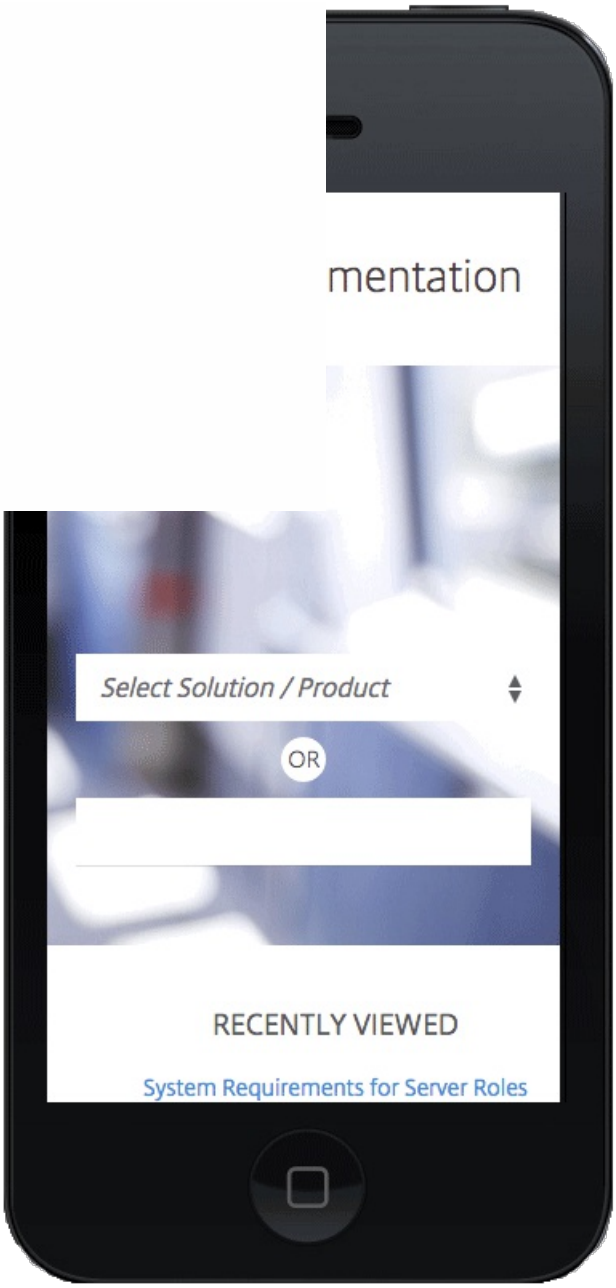
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



RISE Integration: NetScaler ADC and Cisco Nexus 7000 Series Switch

May 19, 2014

Application-delivery solutions that use service modules or switches have required complex configurations and changes in the networking stack. Cisco Remote Integrated Service Engine (RISE) technology can logically integrate a Citrix NetScaler application delivery controller (ADC) with a Cisco Nexus 7000 Series switch as a virtual service module. The integration eliminates the complexities and limitations of traditional inline and one-arm mode configurations. The NetScaler functionality is available as a centralized resource that can be leveraged across the application infrastructure supported by the Cisco Nexus 7000 series switch. The RISE integration provides a service module's streamlined deployment and simplified configuration and operation, and the complete NetScaler application-delivery capabilities that can accelerate application performance for all users.

An ADC deployed in an inline mode can create a bottleneck in the data center. Today, data-center traffic can handle traffic in the terms of terabytes per second, but the ADC capacity can scale up only to gigabytes per second range. The NetScaler ADC, with the new capability to integrate with Cisco Nexus 7000 switch using RISE, can be deployed in a one-arm mode and can provide ADC capabilities to data-center traffic with minimal configuration and maintenance overhead.

A one-arm mode deployment uses Source NAT (SNAT) and Policy Based Routing (PBR) to send only the necessary traffic through the ADC. With SNAT, the servers do not have visibility of client IP addresses. PBR resolves this issue, but it requires complex, manual configuration and is prone to errors. The Automatic Policy Based Routing (APBR) feature in RISE eliminates the need for Source NAT or manual PBR configuration in a one-arm mode design.

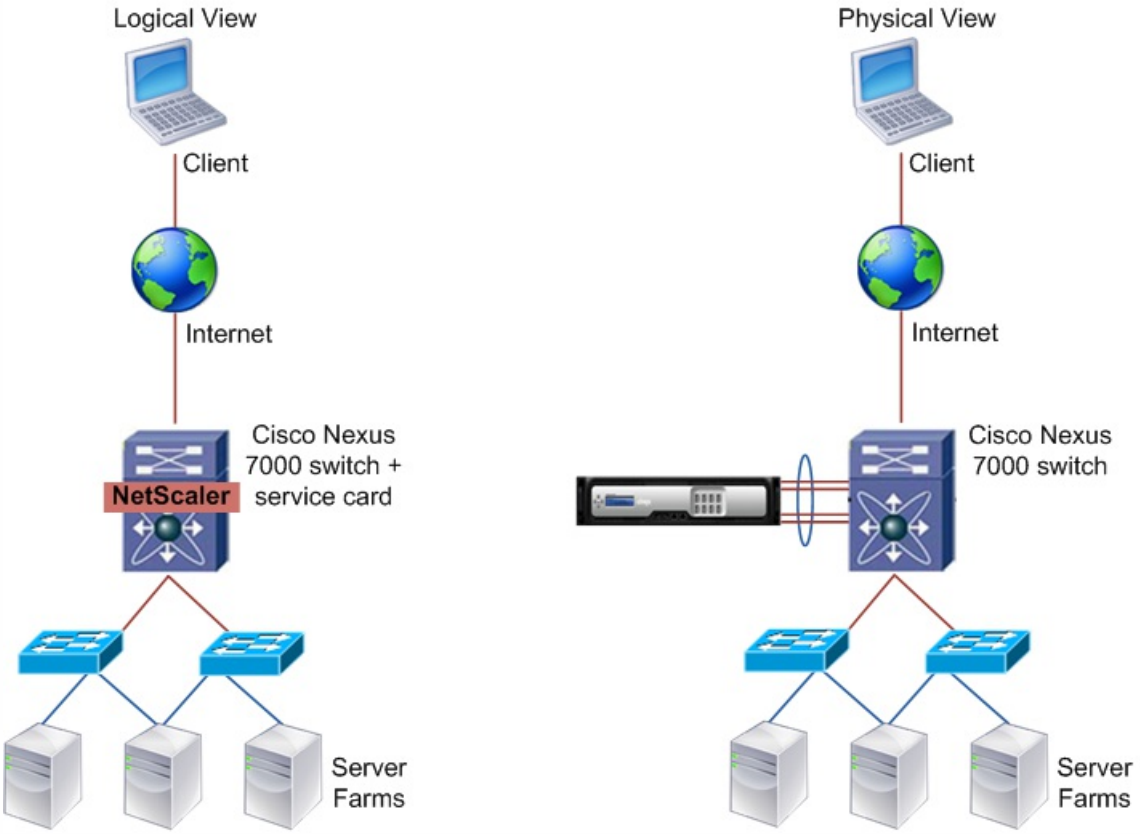
For more information about RISE functionality and other features, see Cisco Remote Integrated Service Engine for Citrix NetScaler Appliances and Cisco Nexus 7000 Series Switches Configuration Guide.

Understanding RISE

May 29, 2014

Cisco RISE technology logically integrates a Citrix NetScaler ADC with a Cisco Nexus 7000 Series switch as a virtual service module. After you connect the NetScaler ADC and the Cisco Nexus 7000 series switch, an initial handshake is performed and a control channel is established between the two devices to exchange port-channel information. The following figure shows the RISE deployment:

Figure 1. RISE Deployment



Because the NetScaler ADC appears to be a virtual module in the switch, client traffic that reaches the Cisco Nexus 7000 series switch is intelligently routed to the NetScaler ADC and then to the servers. The return traffic flows to the ADC through the Cisco switch, and then back to the client.

The interface or port-channel that connects the NetScaler ADC and Cisco Nexus 7000 series switch is a single trunk carrying both control and data VLANs. The control VLAN is used for all control channel communication, and the data VLAN is used for communicating data traffic.

For more information, see Cisco RISE Integration Overview.

This document includes the following:

- RISE Functionality
- RISE Network Topologies
- RISE Connection Modes

RISE Functionality

Updated: 2014-05-19

The feature integration that RISE enables between the NetScaler ADC and the Cisco Nexus 7000 Series switch provides the following functionalities:

- Plug and play auto-provisioning
  RISE provides a plug and play auto-provisioning feature. You can directly connect the NetScaler ADC to the Cisco Nexus 7000 series switch.

- Discovery and bootstrapping
  The discovery and bootstrap mechanism enables the Cisco Nexus 7000 Series switch to perform the initial setup of NetScaler automatically by exchanging information such as NSIP and VLANs to set up a RISE channel, which transmits control and data packets. For details, see Discovery and Bootstrap.

- Health Monitoring
  The NetScaler ADC uses its health monitoring feature to track and support server health by sending health probes to verify server responses. The Intelligent Services Control Manager (iSCM) on the Cisco Nexus 7000 Series switch and the Intelligent Services Control Client (iSCC) on the NetScaler ADC also periodically send heartbeat packets to each other. If a critical error occurs and health monitoring detects a service instance failure, or if the heartbeat is missed six times successively, the RISE channel becomes nonoperational. For details, see Health Monitoring.

- APBR
  Automatic Policy Based Routing (APBR) automatically routes the return traffic from the servers to the NetScaler ADC, preserving the client IP addresses. The automatic policy based routes are defined on the Cisco Nexus 7000 series switch. When the return traffic from the server reaches the Cisco Nexus 7000 series switch, the APBR policies defined on the switch route the traffic to the NetScaler ADC, which in turn routes the traffic to the client.

  Note:
  - APBR can function only if USIP is enabled on the NetScaler ADC.
  - APBR can be deployed in a VPC mode or a non-VPC mode. For more details on VPC mode, see Cisco VPC.

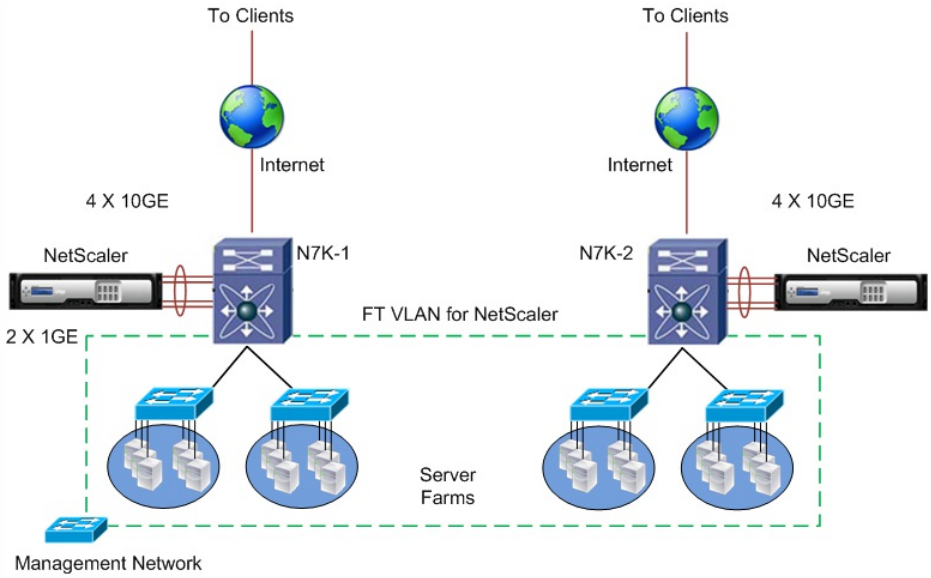
  For details on configuring APBR, see Configuring Auto Policy-Based Routing.

RISE Network Topologies

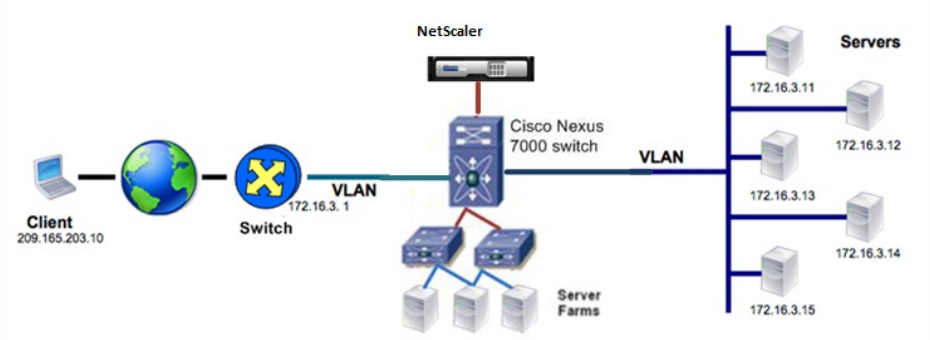
Updated: 2014-05-19

RISE can be deployed in any of the following modes:

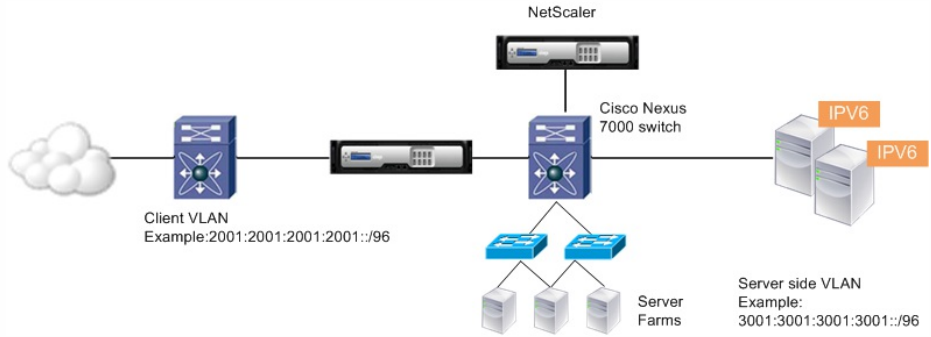
- One-Arm mode— The NetScaler ADC's ports are bundled as a port channel connected to the Cisco Nexus 7000 Series switch. In one-arm mode, the ADC is configured with a VLAN that handles both client and server requests.
  Figure 2. One-Arm Mode



- Bridged mode— In bridged mode, the NetScaler ADC bridges traffic between two VLANs in the same IP subnet. The VLAN facing the WAN is the client VLAN. The VLAN facing the data center is the server VLAN. A bridge group virtual interface (BVI) joins the two VLANs into one bridge group.
  Figure 3. Bridged Mode



- Routed mode—In routed mode, the NetScaler ADC is the next hop in the network, typically with the client-side VLAN and the server-side VLAN in different IP subnets or in different IP networks.
  Figure 4. Routed Mode



RISE Connection Modes

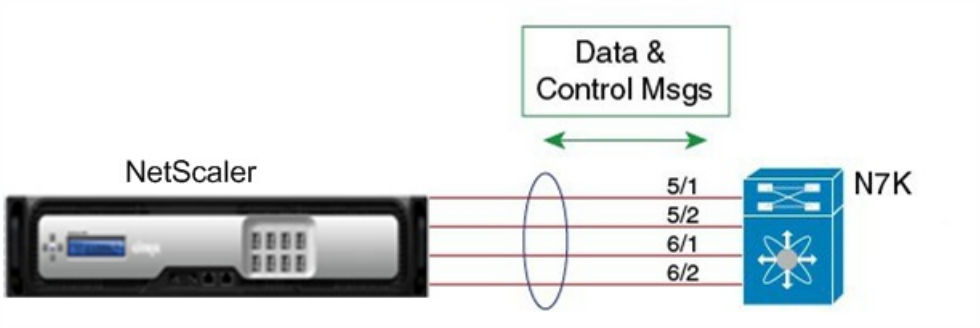
Updated: 2014-05-27

You can connect the Citrix NetScaler appliance to the Cisco Nexus 7000 Series switch in one of the following ways:

Direct Connect Mode for a Standalone Switch

In a direct mode deployment, the NetScaler ADC is attached to a single Nexus 7000 Series switch. The switch can be standalone device or a vPC peer.

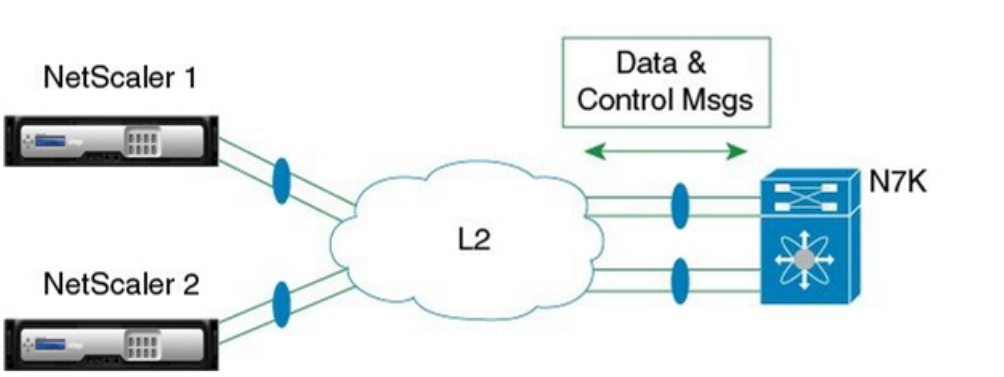
Figure 5. Direct Connect Mode



Indirect Connect Mode

In an indirect mode deployment, a virtual NetScaler ADC is connected to a Cisco Nexus 7000 Series switch through a switched layer 2 network.

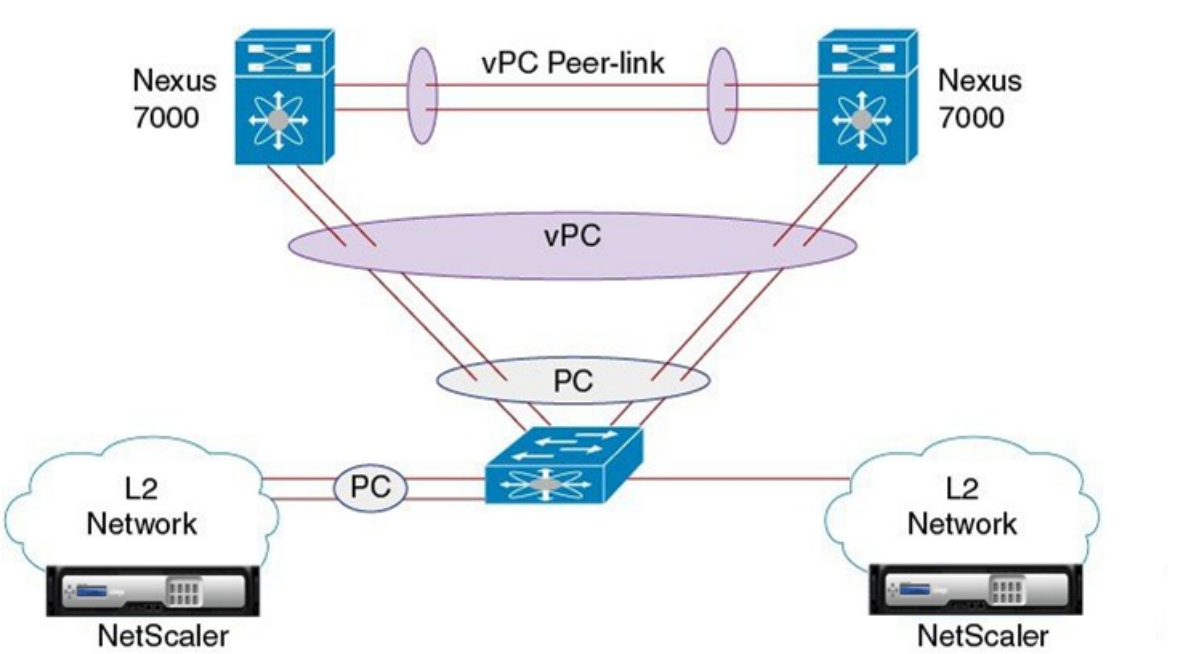
Figure 6. Indirect Connect Mode



Virtual Port Channel (vPC) Connect Mode

In a virtual port channel (vPC) direct mode deployment, the NetScaler ADC is attached to a single Nexus 7000 Series switch that is a vPC peer.

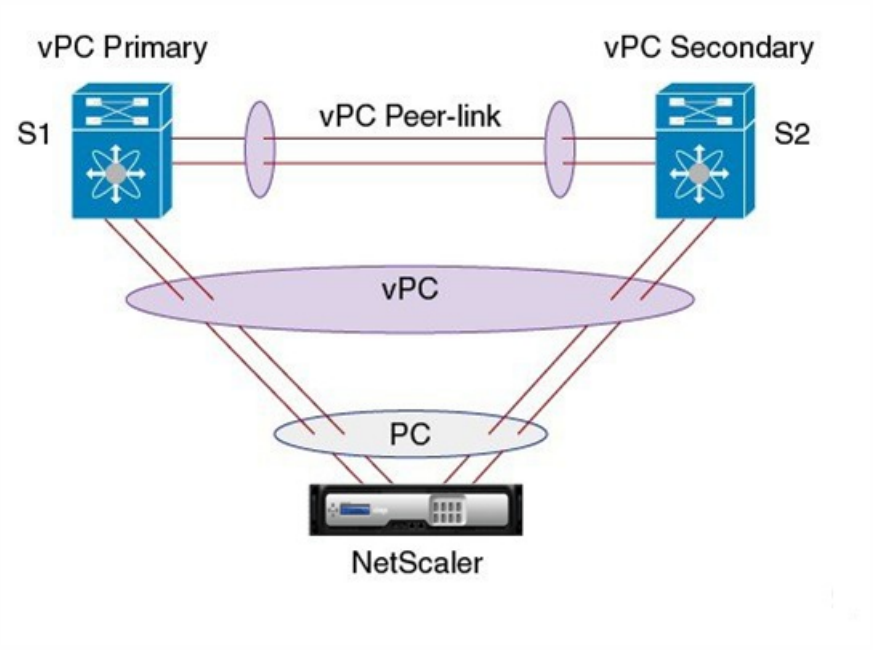
Figure 7. Virtual Port Channel (vPC) Connect Mode



vPC Indirect Connect Mode

In a vPC indirect mode deployment, the NetScaler ADC is indirectly attached to a Cisco Nexus vPC peer through a layer 2 network.

Figure 8. vPC Indirect Connect Mode

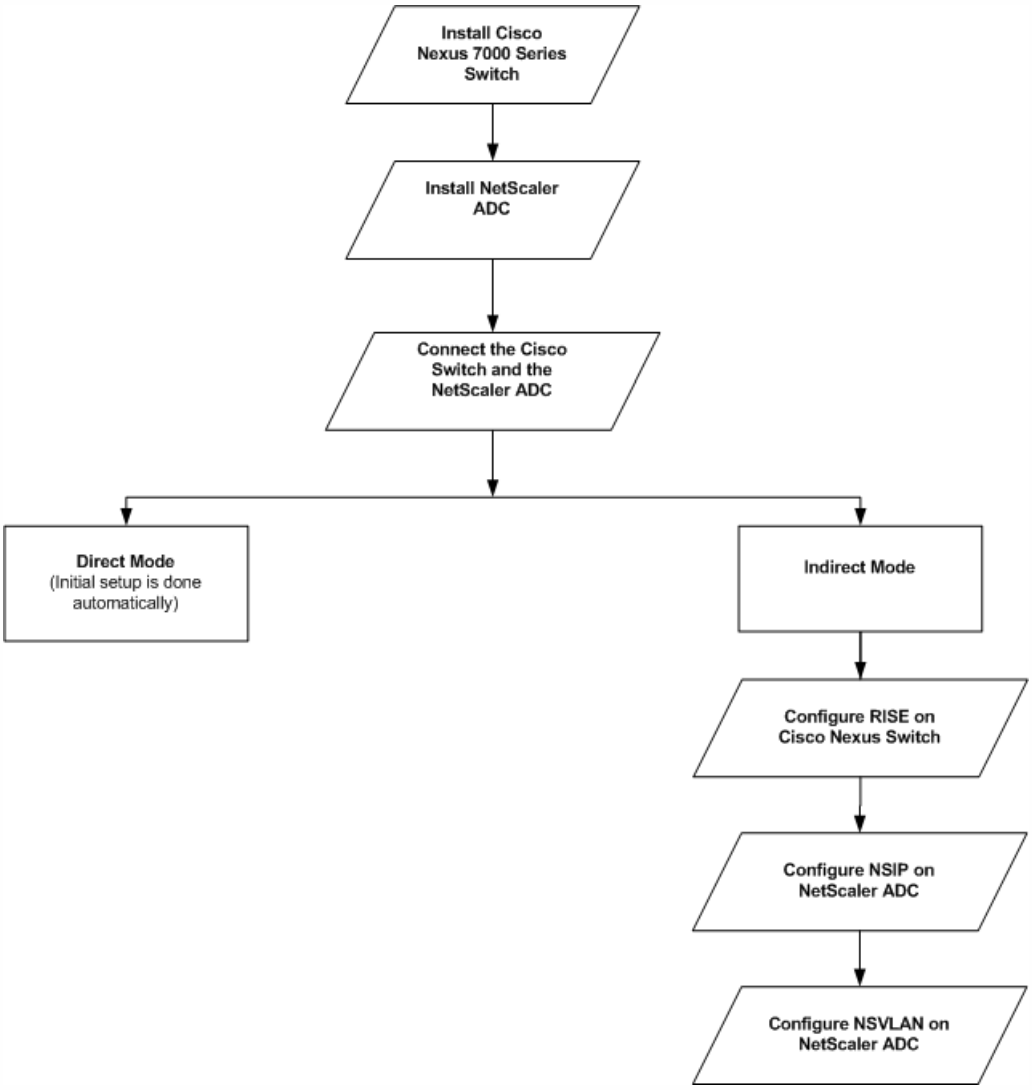


For more information on connection modes, see Connection Modes.

Getting Started with RISE

May 19, 2014

To begin using the RISE features, first install the Cisco Nexus 7000 series switch and the NetScaler ADC. Then, connect the NetScaler ADC to the Cisco Nexus 7000 switch either directly or indirectly. If you plan to connect the NetScaler ADC directly to the Cisco Nexus 7000 switch, the initial setup is done automatically by the auto-discovery feature. The following figure explains the flow of steps in the configuration of RISE.



This document includes the following:

- Installing the Cisco Nexus Switch and the NetScaler ADC
- Accessing the Cisco Nexus Switch and the NetScaler ADC
- Configuring RISE
- Configuring High Availability

Installing the Cisco Nexus Switch and the NetScaler ADC

Updated: 2014-05-26

You need to first install the Cisco Nexus 7000 series switch, and then install the NetScaler ADC.

Installing Cisco Nexus 7000 Series Switch

To install the Cisco Nexus 7000 Series Switch, see Installing Cisco Nexus 7000 Series Switch.

Installing NetScaler ADC

To install the NetScaler ADC, see Installing NetScaler ADC.

To install a virtual NetScaler ADC appliance, see Installing NetScaler Virtual Appliances on XenServer or Installing NetScaler Virtual Appliances on VMware ESX.

Accessing the Cisco Nexus Switch and the NetScaler ADC

Updated: 2014-05-26

After you have completed the installation process, you can access the Cisco Nexus Series 7000 switch through the command-line interface (CLI) and the Citrix NetScaler appliance through the graphical user interface (GUI) or the CLI. To perform administrative tasks and also, to configure RISE, you need to access the Cisco switch and the NetScaler ADC.

Accessing the Cisco Nexus Series 7000 Switch

After installing the Cisco Nexus 7000 series switch, you can access it using the command line interface. To access the Cisco Nexus 7000 Series Switch, see Accessing the Cisco Nexus 7000 series switch.

Accessing the NetScaler ADC

A NetScaler appliance has both a command line interface (CLI) and a graphical user interface (GUI). The GUI includes a configuration utility for configuring the appliance and a statistical utility, called Dashboard. For initial access, all appliances ship with the default NetScaler IP address (NSIP) of 192.168.100.1 and default subnet mask of 255.255.0.0. You can assign a new NSIP and an associated subnet mask during initial configuration.
Note: If you are using the direct connect mode to connect the appliance to the Cisco Nexus switch, you are not required to access the Citrix Netscaler appliance to configure RISE. For direct connect mode, the IP address and VLAN for management are pushed from the Cisco Nexus switch as part of RISE simplified provisioning.
For information about the procedures to access through CLI and GUI, see Using the Command Line Interface and Using Graphical User Interface.

Configuring RISE

Updated: 2015-03-18

After you install the Cisco Nexus 7000 series switch and the NetScaler ADC, configure the appliance to work in direct mode, indirect mode, VPC direct mode, or VPC indirect mode.

Direct Mode

The NetScaler ADC which is directly connected to the switch is automatically configured for RISE mode and all of its ports are in operation mode. No configuration is required on the appliance in a direct mode deployment. For details on configuring RISE in direct mode, see Configuring RISE in Direct Mode.

Indirect Mode

For indirect mode deployment, configure RISE on Cisco Nexus switch, and then configure NSIP and NSVLAN on NetScaler

ADC.

- Configuring RISE on Cisco Nexus Switch
- Configuring NSIP on NetScaler ADC
- Configuring NSVLAN on NetScaler ADC

If the direct mode is disabled. all Layer 2 (L2) discovery messages are dropped and L2 RISE discovery does not take place. While, if the direct mode is enabled, the L2 discovery messages are parsed. If L2 RISE discovery messages are successfully parsed, the Indirect Mode is automatically enabled.

If indirect mode is disabled, NetScaler RISE daemon does not listen on TCP ports 8000 and 8001. As a result, RISE heartbeats stop and RISE profiles go down. While, if indirect mode is enabled, NetScaler RISE daemon starts listening on TCP ports 8000 and 8001. As a result, RISE heartbeats may be exchanged and RISE profiles may come up.

The default setting for RISE is direct attach mode as enabled and indirect mode as disabled. With these settings, the direct attach mode works. However, indirect mode will not work. If you want to deploy NetScaler in an indirect mode setting with N7K then use the following setting:

set rise param -indirectMode ENABLED

With active RISE profiles in direct mode, you must consider the followings points:

1. The set rise param -directMode DISABLED command displays a warning message; Warning: Disabling direct mode with one or more active RISE profile(s) on DIRECT ATTACH mode may cause corresponding RISE profile(s) to not come UP on reboot
2. The set rise param -indirectMode DISABLED displays a warning message; Warning: Disabling indirect mode will cause any active RISE profile(s) to go down

However, if the direct mode is enabled, and NetScaler starts receiving L2 RISE discovery messages, it automatically turns the indirect mode as enabled. Otherwise, RISE direct mode does not work.

With active RISE profiles in Indirect Attach mode, the set rise param -indirectMode DISABLED command displays a warning message; Warning: Disabling indirect mode will cause any active RISE profile(s) to go down leading to RISE profiles on Indirect Attach going down.

vPC Direct Mode

The NetScaler ADC which is directly connected to the switch is automatically configured for RISE mode and all of its ports are in operation mode. No configuration is required on the appliance in a direct mode deployment. For details on configuring RISE in direct mode, see Configuring RISE in vPC Direct Mode.

vPC Indirect Mode

In a vPC indirect mode deployment, the NetScaler ADC is indirectly attached to a Cisco Nexus vPC peer through a Layer 2 network.

- Configuring RISE on Cisco Nexus Switch
- Configuring NSIP on NetScaler ADC
- Configuring NSVLAN on NetScaler ADC

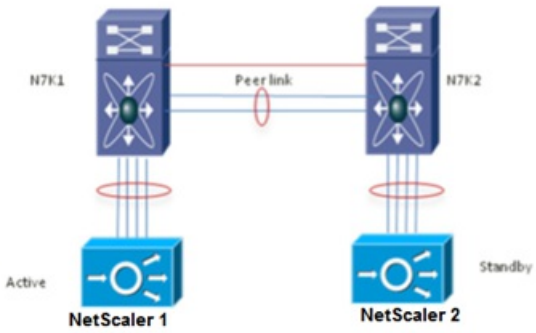
For details on configuring RISE in direct mode, see Configuring RISE in vPC Direct Mode.
For information on configuring RISE, see Configuring RISE.

Configuring High Availability

Updated: 2014-05-26

In a High Availability setup, the RISE deployment uses a maximum of two NetScaler ADCs. If one ADC becomes unavailable, the traffic flow is seamlessly switched to the other ADC.

Figure 1. High Availability



For information about high availability configuration, see High Availability.

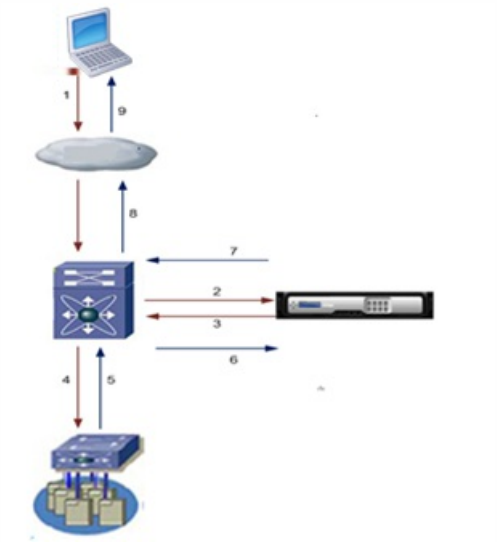
Use Case: Configuring Auto Policy-Based Routing

May 27, 2014

Auto Policy-Based Routing (APBR) automatically routes the return traffic from the servers to the NetScaler ADC, preserving the client IP addresses. The automatic policy based routes are defined on the Cisco Nexus 7000 series switch. When the return traffic from the server reaches the Cisco Nexus 7000 series switch, the APBR policies defined on the switch route the traffic to the NetScaler ADC, which in turn routes the traffic to the client.

To understand the need for APBR, first consider a NAT based scenario in which a packet flows from the client to the server and from the server back to the client.

Figure 1. Packet Flow



1. Client initiates the traffic to the virtual IP (VIP) address.
   SRC_IP= Client IP; DST_IP= VIP
2. The Cisco Nexus switch forwards the packet to the NetScaler ADC.
   SRC_IP= Client IP; DST_IP= VIP
3. The ADC performs source NAT and destination NAT (Network Address Translation), changes the source IP and destination IP addresses, and sends the packet to the Cisco Nexus switch.
   SRC_IP= NAT_IP; DST_IP= RS_IP
4. The Cisco Nexus switch receives the packet and forwards it to a server.
   SRC_IP= NAT_IP; DST_IP= RS_IP
5. The server processes the packet and forwards it to the Cisco Nexus 7000 series switch.
   SRC_IP= RS_IP IP; DST_IP= NAT_IP
6. The Cisco Nexus switch forwards the packet to the NetScaler ADC.
   SRC_IP= RS_IP IP; DST_IP= NAT_IP
7. The NetScaler ADC changes the source IP address and forwards the packet to the Cisco Nexus 7000 series switch.
   SRC_IP= VIP; DST_IP= Client_IP

8. The Cisco Nexus 7000 series switch forwards the packet to the client.
   SRC_IP= VIP; DST_IP= Client_IP

   The client receives the packet. However, the client IP address is not visible to the server.

Now, consider a scenario in which policy based routing (PBR) directs packet flow.

1. Client initiates the traffic to the virtual IP (VIP) address.
   SRC_IP= Client IP; DST_IP= VIP
2. The Cisco Nexus switch forwards the packet to the NetScaler ADC.
   SRC_IP= Client IP; DST_IP= VIP
3. The ADC performs destination NAT (Network Address Translation), changes the destination IP, and then sends the packet to the Cisco Nexus switch.
   SRC_IP= Client IP; DST_IP= RS_IP
4. The Cisco Nexus switch receives the packet and forwards it to a server.
   SRC_IP= Client IP; DST_IP= RS_IP
5. The server processes the packet and forwards it to the Cisco Nexus 7000 series switch.
   SRC_IP= RS_IP IP; DST_IP= Client IP
6. The Cisco Nexus switch forwards the packet to the NetScaler ADC.
   SRC_IP= RS_IP IP; DST_IP= Client IP
7. The NetScaler ADC changes the source IP address and forwards the packet to the Cisco Nexus 7000 series switch.
   SRC_IP= VIP; DST_IP= Client_IP
8. The Cisco Nexus 7000 series switch forwards the packet to the client.
   SRC_IP= VIP; DST_IP= Client_IP
9. The client receives the packet. The client IP address is visible to the server. However, PBR requires manual and complex configurations and is prone to errors.

To overcome these drawbacks, configure APBR rules on the RISE appliance. When APBR is configured, the packets flow as described in the following procedure:

1. Client initiates the traffic to the virtual IP (VIP) address.
   SRC_IP= Client IP; DST_IP= VIP
2. The Cisco Nexus switch forwards the packet to the NetScaler ADC.
   SRC_IP= Client IP; DST_IP= VIP
3. The ADC performs load balancing and changes the destination IP address to the appropriate server IP address and forwards the packet to the Cisco Nexus switch in an APBR message.
   SRC_IP= Client IP; DST_IP= RS_IP
4. The Cisco Nexus switch receives the packet and forwards it to a server by using a route map.
   SRC_IP= Client IP; DST_IP= RS_IP
5. The server processes the packet and forwards it to the Cisco Nexus 7000 series switch.
   SRC_IP= RS_IP IP; DST_IP= Client_IP

6. When the packet reaches the Nexus switch, the switch applies the APBR rules, sets the next hop IP address to that of the NetScaler ADC, and forwards the packet to the NetScaler ADC.
   SRC_IP= RS_IP IP; DST_IP= Client_IP

7. The NetScaler ADC changes the source IP address and forwards the packet to the Cisco Nexus 7000 series switch.
   SRC_IP= VIP; DST_IP= Client_IP

8. The Cisco Nexus 7000 series switch forwards the packet to the client.
   SRC_IP= VIP; DST_IP= Client_IP

9. The client receives the packet successfully.

Note: APBR rules are configured on the Cisco Nexus switch by the Citrix Netscaler appliance only if the Use Source IP (USIP) option is enabled in the services or service groups on the Citrix Netscaler appliance.
The APBR message control flow is explained below

1. After USIP is enabled in the services on Netscaler ADC, it publishes the IP address, port number and protocol details of the server to the Cisco Nexus 7000 series switch over the RISE control channel.
2. Using the IP address, port number and protocol details of the server, the Cisco Nexus 7000 series switch creates an APBR rule which consists of ACLs and route maps.
   Note:
   - For local servers, the switch creates ACLs and route maps.
   - For remote servers , the switch forwards the APBR messages to other Cisco Nexus 7000 series switches.
3. The RISE appliance then applies the APBR rules to the switch virtual interface on the Cisco Nexus 7000 series switch connected to server.

To configure the APBR functionality:

- Enable the feature on the Cisco Nexus switch
- Configure APBR on NetScaler ADC
  - Configure NSIP
  - Configure NSVLAN
  - Enable USIP option

For more information, see Configuring Auto Policy-Based Routing.

Web Interface

Aug 30, 2013

The Web Interface on Citrix NetScaler appliances is based on Java Server Pages (JSP) technology and provides access to Citrix XenApp and Citrix XenDesktop applications. Users access resources through a standard Web browser or by using the Citrix XenApp plug-in.

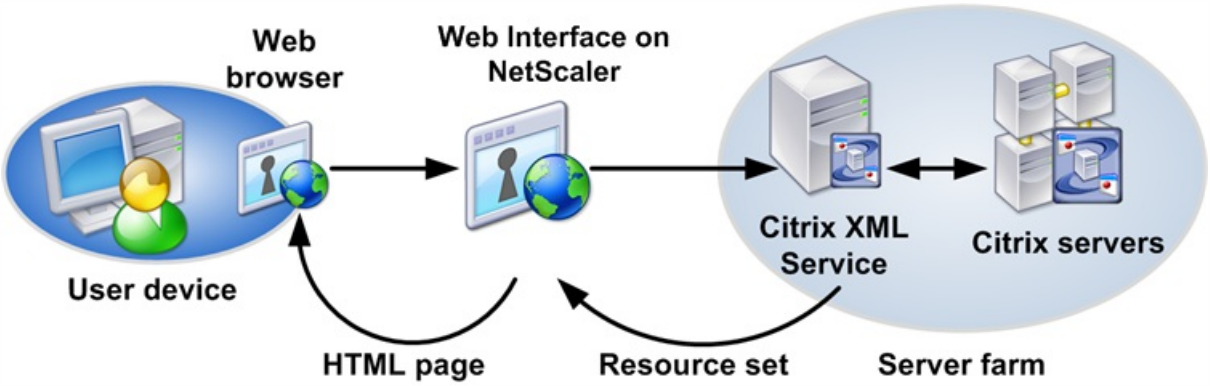
The Web Interface runs as a service on port 8080 on the NetScaler appliance. To create Web Interface sites, Java is executed on Apache Tomcat Web server version 6.0.35 on the NetScaler appliance. The Web Interface sites provide user access to the XenApp and XenDesktop resources, which include applications, content, and desktops.

The Web Interface installation includes installing the Web Interface tar file and JRE tar file on the NetScaler appliance. To configure the Web Interface, you create a Web Interface site and bind one or more XenApp or XenDesktop farms to it.

How Web Interface Works

The following figure illustrates a basic Web interface session.

Figure 1. *A Basic Web Interface Session*



Following is a typical set of interactions among a user device, a NetScaler running the Web interface, and a server farm.

1. A user authenticates to the Web interface through a Web browser or by using the XenApp plug-in.
2. The Web interface reads the user's credentials and forwards the information to the Citrix XML Service running on servers in the server farm.
3. The Citrix XML Service on the designated server retrieves from the servers a list of resources that the user can access. These resources constitute the user's resource set and are retrieved from the Independent Management Architecture (IMA) system.
4. The Citrix XML Service then returns the user's resource set to the Web interface running on the NetScaler.
5. The user clicks an icon that represents a resource on the HTML page.
6. The Web interface queries the Citrix XML Service for the least busy server.
7. The Citrix XML Service returns the address of this server to the Web interface.
8. The Web interface sends the connection information to the Web browser.
9. The Web browser initiates a session with the server.

Installing and Configuring the Web Interface

Sep 27, 2013

To configure the web interface, you create a web interface site and bind one or more XenApp or XenDesktop farms to it. You then configure the web interface to work behind an HTTP or an HTTPS virtual server or NetScaler Gateway.

- **Using an HTTP or an HTTPS virtual server**. You create an HTTP or an HTTPS virtual server on the NetScaler appliance and bind the web interface service, running on port 8080 of the NetScaler appliance, to the virtual server. Clients on the LAN use the virtual server IP address to access the web interface. When using this access method, the URL format for the web interface site is as follows:

  <HTTP or HTTPS>://<HTTP or HTTPS vserver IP address>:<vserver port number>/<web interface site path>

  The following access methods are available for clients accessing the web interface site when it is configured using an HTTP or an HTTPS virtual server:
  - **Direct**. Actual address of a XenApp or XenDesktop server is sent to the clients.
  - **Alternate**. Alternate address of a XenApp or XenDesktop server is sent to the clients.
  - **Translated**. Translated address, from the defined internal addresses to external addresses and ports mapping table, is sent to the clients from a specified network. When you use this option, you have to define internal address to external address and port mappings.
- **Using NetScaler Gateway**. You associate the web interface site with NetScaler Gateway. You associate the web interface site with NetScaler Gateway. Remote clients use the NetScaler Gateway URL to access the web interface site. With this access method, the URL format for the web interface site is as follows:

  HTTPS://<Access Gateway URL>/<web interface site path>

  The following access methods are available for clients accessing the web interface site when it is configured using an NetScaler Gateway:
  - **Gateway Direct**. Actual address of a XenApp or XenDesktop server is sent to NetScaler Gateway.
  - **Gateway Alternate**. Alternate address of a XenApp server is sent to NetScaler Gateway. You cannot use this mode to access XenDesktop servers.
  - **Gateway Translated**. Translated address, from the defined internal addresses to external addresses and ports mapping table, is sent to NetScaler Gateway. When you use this option, you have to define internal address to external address and port mappings.

Prerequisites

Updated: 2013-09-06

The following prerequisites are required before you begin installing and configuring the Web interface.

- XenApp or XenDesktop farms are set up and running in your environment. For more information about XenApp, see the XenApp documentation at http://edocs.citrix.com/. For more information about XenDesktop, see the XenDesktop farms documentation at http://edocs.citrix.com/.
- Conceptual knowledge of the Web interface. For more information about Web interface running on a server, see the Web interface documentation at http://edocs.citrix.com/.

Installing the Web Interface

To install the Web interface, you need to install the following files:

- **Web interface tar file**. The setup file for installing the Web interface on the NetScaler appliance. This tar file also includes Apache Tomcat Web server version 6.0.35. The file name has the following format: nswi-<version number>.tgz (for example, nswi-1.5.tgz).
- **JRE tar file**. The JRE tarball. You can use the Diablo Latte JRE version 1.6.0-7 for 64-bit FreeBSD 6.x/amd64 platform available on FreeBSD Foundation Web site at http://www.freebsdfoundation.org/java/java16. Alternatively, you can use OpenJDK6 package for FreeBSD 6.x/amd63. You can download openjdk6-b17_2.tbz from https://citrix.sharefile.com/d/c85aeefcc05643f8.

Note: On a high availability setup, when installing the web interface with tar files (web interface and JRE) that are already available on the appliance, ensure that the files are available in the same location on both the primary and secondary appliances; otherwise, the web interface will not be installed on the secondary appliance.
Copy the tar files to a local workstation or to the /var directory of the appliance.

These files install all the Web interface components and JRE on the hard drive and configure automatic startup of the Tomcat Web server with Web interface at appliance startup time. Both tar files are internally expanded in the /var/wi directory on the hard drive.

Note: After installing web interface on the appliance and before creating a web interface site, you must place the client plugin in the appliance by using the appropriate Upload Plugins utility provided on the web interface details pane.

To install the Web interface and JRE tar files by using the command line interface

At the command prompt, type:

install wi package -wi <URL> -jre <URL>

Examples

> install wi package -wi sftp://username:password@10.102.29.12/var/ nswi-1.5.tgz -jre <url> > install wi package -wi ftp://username:password@10.102.29.15/var/ nswi-

To install the Web interface and JRE tar files by using the configuration utility

1. Navigate to System > Web Interface and, in the Getting Started group, click Install Web Interface.
2. Specify the install path for web interface tar file and JRE tar file.

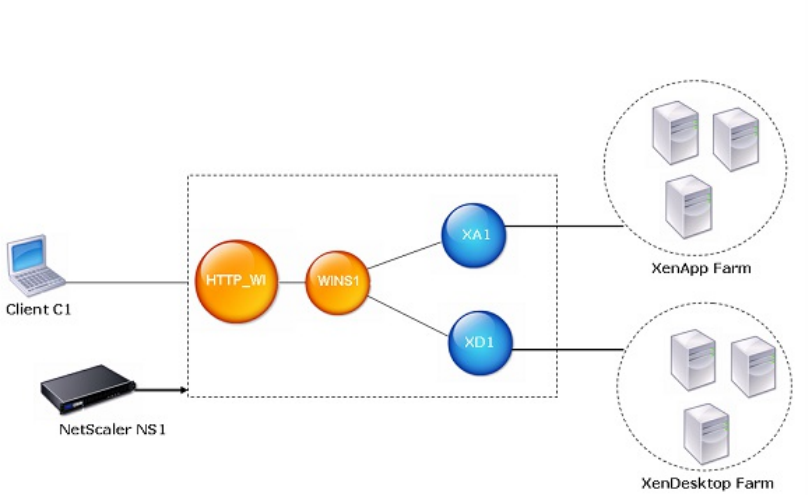
Configuring a Web Interface Site for LAN Users Using HTTP

Aug 08, 2014

In this scenario, user and the Web interface setup are on the same enterprise LAN. The enterprise has both a XenApp and a XenDesktop farm. Users access the Web interface by using an HTTP vserver. The Web interface exposes its own login page for authentication. The vserver IP address is used to access the Web interface.

The following figure illustrates the Web interface running on the NetScaler appliance NS1. A Web interface site WINS1 is created and a XenApp farm XA1 and a XenDesktop farm XD1 are bound to it. An HTTP vserver HTTP_WI is also created. Client C1 uses the IP address of the HTTP_WI vserver to access the WINS1 site.

Figure 1. A Web Interface Site Configured for LAN Users Using HTTP



To configure a Web interface site for LAN users using HTTP by using the configuration utility

1. Navigate to System > Web Interface, click Web Interface Wizard, and configure the web interface parameters.
2. In Default Access Methods, select the required access option and configure the access method parameters.
   Note: When you create the HTTP vserver by using the configuration utility, the configuration utility automatically creates a service, which logically represents the Web interface service running on the NetScaler appliance, and binds the service to the HTTP virtual server.
3. On Configure Access Methods page, create the access method for a client IP address or network.
   Note: Before you configure access method based on the client IP address, you must enable USIP mode on the web interface service to make the client's IP address available with the web interface.
4. On Configure Address Translations page, create the address translation for mapping between an Internal IP address and an external IP address.
   Note: The Configure Address Translations page appears on the wizard when you set the Translated access method for a Client's IP address or network.
5. On Configure XenApp/XenDesktop Farm page, create the XenApp or XenDesktop farm.
6. Verify the Web interface configuration by viewing the Details section at the bottom of the pane.
   To view the Web interface site, Navigate toSystem > Web Interface > Sites.

To configure a Web interface site for LAN users using HTTP by using the command line interface

1. Add a Web interface site. Set Direct or Alternate or Translated for the defaultAccessMethod parameter. At the command prompt, type: add wi site <sitePath> -siteType (XenAppWeb | XenAppServices) -publishedResourceType (Online | Offline | DualMode) -kioskMode (ON | OFF) -wiAuthenticationMethods (Explicit | Anonymous) -webSessionTimeout <positive_integer> -defaultAccessMethod <defaultAccessMethod> -loginTitle <string>

Example

```
> add wi site WINS1 -siteType XenAppWeb -publishedResourceType Online -kioskMode ON -defaultAccessMethod Direct
```

2. (Optional) Set an access method for a Client's IP address or network. At the command prompt, type:
   bind wi site <sitePath> -accessMethod <accessMethod> -clientIpAddress <ip_addr> -clientNetMask <netmask>

3. If you have set the Translated access method for a Client's IP address or network then provide Internal IP and external IP address mappings. At the command prompt, type:
   bind wi site <sitePath> -translationInternalIp <ip_addr> -translationInternalPort <port | *> -translationExternalIp <ip_addr> -translationExternalPort <port | *> [-accessType <accessType>]

4. Bind XenApp or XenDesktop farms to the Web interface site. At the command prompt, type:
   bind wi site <sitePath> <farmName> <xmlServerAddresses> -xmlPort <value> -transport (HTTP | HTTPS) -loadBalance (ON | OFF)

   Example

   ```
   > bind wi site WINS1 XA1 10.102.46.6 -xmlPort 80 -transport HTTP -LoadBalance OFF
   > bind wi site WINS1 XD1 10.102.46.50 -xmlPort 80 -transport HTTP -LoadBalance OFF
   ```

5. Create a service that is a logical representation of the Web interface service running on the NetScaler appliance. At the command prompt, type:
   add service <name> <IP address> <serviceType> <port>

   Example

   ```
   > add service WI_Loopback_Service 127.0.0.1 HTTP 8080
   ```

6. Add an HTTP vserver. At the command prompt, type:
   add lb vserver <virtualServerName> <protocol> <IPAddress> <port>

   Example

   ```
   > add lb vserver HTTP_WI HTTP 10.102.29.5 80
   ```

7. Bind the Web interface service to the HTTP vserver. At the command prompt, type:
   bind lb vserver <virtualServerName> <serviceName>

   Example

   ```
   > bind lb vserver HTTP_WI WI_Loopback_Service
   ```

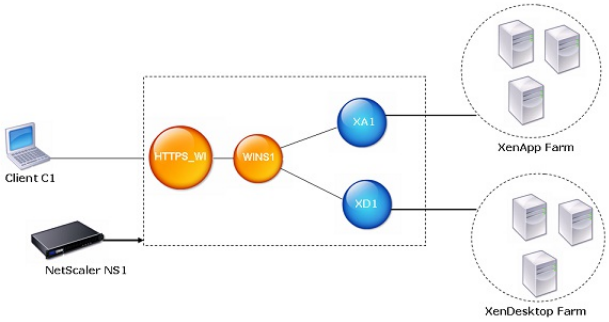
Configuring a Web Interface Site for LAN Users Using HTTPS

Aug 08, 2014

In this scenario, user accounts and the Web interface setup are on the same enterprise LAN. Users access the Web interface by using an SSL-based (HTTPS) vserver. The Web interface exposes its own login page for authentication. SSL offloading is done by this vserver on the NetScaler. The vserver IP address is used to access the Web interface instead of the NetScaler IP address (NSIP).

The following figure illustrates the Web interface running on the NetScaler appliance NS1. A Web interface site WINS1 is created and a XenApp farm XA1 and a XenDesktop farm XD1 are bound to it. An HTTPS vserver HTTPS_WI is also created. Client C1 uses the IP address of the HTTPS_WI vserver to access the WINS1 site.

Figure 1. A Web Interface Site Configured for LAN Users Using HTTPS



To configure a Web interface site for LAN users using HTTPS by using the configuration utility

1. Navigate toSystem > Web Interface, click Web Interface Wizard, and configure the web interface parameters.
2. In Default Access Methods, select the required access option and configure the access method parameters.
   Note: When you create the HTTPS vserver by using the configuration utility, the configuration utility automatically creates a service, which logically represents the Web interface service running on the NetScaler appliance, and binds the service to the HTTPS virtual server.
3. On Specify a server Certificate page, create or specify an existing SSL certificate–key pair. The SSL certificate–key pair is automatically bound to the HTTPS vserver.
4. On Configure Access Methods page, create the access method for a client IP address or network.
   Note: Before you configure access method based on the client IP address, you must enable USIP mode on the web interface service to make the client's IP address available with the web interface.
5. On Configure Address Translations page, create the address translation for mapping between an Internal IP address and an external IP address.
   Note: The Configure Address Translations page appears on the wizard when you set the Translated access method for a Client's IP address or network.
6. On the wizard's Configure XenApp/XenDesktop Farm page, create the XenApp or XenDesktop farm.
7. Verify the Web interface configuration by viewing the Details section at the bottom of the pane.
   To view the Web interface site, Navigate to System > Web Interface > Sites.

To configure a Web interface site for LAN users using HTTPS by using the command line

1. Add a Web interface site. Set Direct or Alternate or Translated for the defaultAccessMethod parameter. At the command prompt, type:
   add wi site <sitePath> -siteType (XenAppWeb | XenAppServices) -publishedResourceType (Online | Offline | DualMode) -kioskMode (ON | OFF) -wiAuthenticationMethods (Explicit | Anonymous) -webSessionTimeout <positive_integer> -defaultAccessMethod <defaultAccessMethod> -loginTitle <string>

   Example
   > add wi site WINS1 -siteType XenAppWeb -publishedResourceType Online -kioskMode ON -defaultAccessMethod Direct
2. (Optional) Set an access method for a Client's IP address or network. At the command prompt, type:
   bind wi site <sitePath> -accessMethod <accessMethod> -clientIpAddress <ip_addr> -clientNetMask <netmask>
3. If you have set the Translated access method for a Client's IP address or network then provide Internal IP and external IP address mappings. At the command prompt, type:
   bind wi site <sitePath> -translationInternalIp <ip_addr> -translationInternalPort <port | *> -translationExternalIp <ip_addr> -translationExternalPort <port | *> [-accessType <accessType>]
4. Bind XenApp or XenDesktop farms to the Web interface site. At the command prompt, type:
   bind wi site <sitePath> <farmName> <xmlServerAddresses> -xmlPort <value> -transport (HTTP | HTTPS) -loadBalance (ON | OFF)

   Example
   > bind wi site WINS1 XA1 10.102.46.6 -xmlPort 80 -transport HTTP -LoadBalance OFF
   > bind wi site WINS1 XD1 10.102.46.50 -xmlPort 80 -transport HTTP -LoadBalance OFF
5. Create a service that is a logical representation of the Web interface service running on the NetScaler appliance. At the command prompt, type:
   add service <name> <IPAddress> <serviceType> <port>

   Example
   > add service WI_Loopback_Service 127.0.0.1 HTTP 8080
6. Add an HTTPS vserver. At the command prompt, type:

add lb vserver <name>@ <protocol> <IPAddress> <port>

Example

```
> add lb vserver HTTPS_WI SSL 10.102.29.3 443
```

7. Bind the Web interface service to the HTTPS vserver. At the command prompt, type:
   bind lb vserver <name>@ <serviceName>

   Example

   ```
   > bind lb vserver HTTPS_WI WI_Loopback_Service
   ```

8. Create an SSL certificate key pair. At the command prompt, type:
   add ssl certkey <certificate-KeyPairName> -cert <certificateFileName> -key <privateKeyFileName>

   Example

   ```
   > add ssl certkey SSL-Certkey-1 -cert /nsconfig/ssl/test1.cer -key /nsconfig/ssl/test1
   ```

9. Bind the SSL certificate key pair to the HTTPS vserver. At the command prompt, type:
   bind ssl vserver <vserverName> -certkeyName <certificate- KeyPairName>

   Example

   ```
   > bind ssl vserver HTTPS_WI -certkeyName SSL-Certkey-1
   ```

10. Add a rewrite action. At the command prompt, type:
    add rewrite action <name> <type> <target> [<stringBuilderExpr>] [(-pattern <expression>]

    Example

    ```
    > add rewrite action Replace_HTTP_to_HTTPS INSERT_AFTER "HTTP.RES.HEADER(\"Location\").Value(0).Prefix(4)" "\"s\""
    ```

11. Create a rewrite policy and bind the rewrite action to it. At the command prompt, type:
    add rewrite policy <name> <rule> <action>

    Example

    ```
    > add rewrite policy rewrite_location "HTTP.RES.STATUS == 302 && HTTP.RES.HEADER(\"Location\").Value(0).startswith(\"http:\")" Replace_HTTP_to_HTTPS
    ```

12. Bind the rewrite policy to the HTTPS vserver. At the command prompt, type:
    bind lb vserver <VserverName> -policyname <rewritePolicyName> -priority <value> -type response

    Example

    ```
    > bind lb vserver HTTPS_WI -policyname rewrite_location -priority 10 -type response
    ```

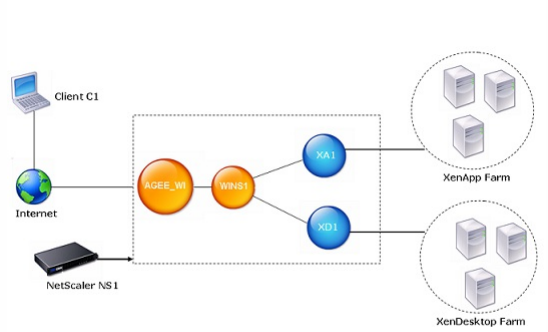
Configuring a Web Interface Site for Remote Users Using NetScaler Gateway

Aug 08, 2014

In this scenario, user accounts and the Web interface setup are on different networks. Users access a Web interface site by using the NetScaler Gateway URL. SmartAccess is automatically enabled.

The following figure illustrates the Web interface running on the NetScaler appliance NS1. A Web interface site WINS1 is created and a XenApp farm XA1 and a XenDesktop XD1 are bound to it. An NetScaler Gateway VPN vserver AGEE_WI is also configured. The client uses the NetScaler Gateway URL of the AGEE_WI to access the WINS1 site.

Figure 1. A Web Interface Site Configured for Remote Users Using NetScaler Gateway



To configure a Web interface site for remote users using NetScaler Gateway by using the configuration utility

1. Navigate to System > Web Interface, click Web Interface Wizard, and configure the web interface parameters.
2. In Default Access Methods, select the required access option and configure the access method parameters.
3. On Configure Access Methods page, create the access method for a client IP address or network.
   Note: Before you configure access method based on the client IP address, you must enable USIP mode on the web interface service to make the client's IP address available with the web interface.
4. On Configure Address Translations page, create the address translation for mapping between an Internal IP address and an external IP address.
   Note: The Configure Address Translations page appears on the wizard when you set the Translated access method for a Client's IP address or network.
5. On Configure XenApp/XenDesktop Farm page, create the XenApp or XenDesktop farm.
6. Verify the Web interface configuration by viewing the Details section at the bottom of the pane.
   To view the Web interface site, Navigate to System > Web Interface > Sites.

To configure a Web interface site for remote users using NetScaler Gateway by using the command line interface

1. Add a Web interface site. Set GatewayDirect or GatewayAlternate or GatewayTranslated for the defaultAccessMethod parameter. At the command prompt, type:
   add wi site <sitePath> <agURL> <staURL> -sessionReliability (ON | OFF) -useTwoTickets (ON | OFF) -secondSTAURL <string> -authenticationPoint (WebInterface | AccessGateway) -siteType (XenAppWeb | XenAppServices) -publishedResourceType (Online | Offline | DualMode) -kioskMode (ON | OFF)-wiAuthenticationMethods (Explicit | Anonymous) -webSessionTimeout <positive_integer> -defaultAccessMethod <defaultAccessMethod> -loginTitle <string>

   Example
   > add wi site WINS1 https://ag.mycompany.com http://ag.staserver.com -sessionReliability OFF -authenticationPoint AccessGateway -siteType XenAppWeb -publishedRes
2. (Optional) Set an access method for a Client's IP address or network. At the command prompt, type:
   bind wi site <sitePath> -accessMethod <accessMethod> -clientIpAddress <ip_addr> -clientNetMask <netmask>
3. If you have set the Translated access method for a Client's IP address or network then provide Internal IP and external IP address mappings. At the command prompt, type:
   bind wi site <sitePath> -translationInternalIp <ip_addr> -translationInternalPort <port | *> -translationExternalIp <ip_addr> -translationExternalPort <port | *> [-accessType <accessType>]
4. Bind XenApp or XenDesktop farms to the Web interface site. At the command prompt, type:
   bind wi site <sitePath> <farmName> <xmlServerAddresses> -xmlPort <value> -transport (HTTP | HTTPS) -loadBalance (ON | OFF)

   Example
   > bind wi site WINS1 XA1 10.102.46.6 -xmlPort 80 -transport HTTP -LoadBalance OFF
   > bind wi site WINS1 XD1 10.102.46.50 -xmlPort 80 -transport HTTP -LoadBalance OFF

Using Smart Card Authentication for Web Interface through NetScaler Gateway

Aug 08, 2014

The web interface on the NetScaler appliance supports single sign-on with a smart card through NetScaler Gateway. You log on to NetScaler Gateway by using a valid client certificate, either served from the local certificate store or from a smart card. After successful authentication, you are redirected to the web interface.

Requirements

- Make sure that you install the latest web interface tar file (nswi-1.5.tgz). This tar file provides support for smart card authentication.
- You must configure Delegation on the Active Directory. Follow the **Active Directory Configuration** section under **Procedure**, as described in the article at http://support.citrix.com/article/CTX124603.

To use smart card authentication for a web interface site through NetScaler Gateway by using the configuration utility

1. Navigate to System > Web Interface, click Web Interface Wizard, and configure the web interface parameters.
2. Select GatewayDirect as the Default Access Method.
3. Select Access Gateway as the Authentication Point.
4. Create theAccess Gateway Vserver. For more information, see "http://support.citrix.com/proddocs/topic/access-gateway-10/agee-config-settings-ag-wizard-tsk.html."
5. Click Settings to set the ICA mode, SSO, and WIhome and select SmartCard as the Access Gateway Authentication Method.
6. Click the SmartCard Settings link to configure certificate-based authentication on Access Gateway. You can skip these steps if certificate-based authentication is already configured on Access Gateway. In the Smart Card Settings wizard do the following:
    1. Specify the CA certificate. Install a CA certificate or use an already installed certificate for the NetScaler Gateway virtual server to authenticate the client certificate.
    2. Configure authentication policies. Create a certificate-based authentication policy and bind it to NetScaler Gateway vserver as follows:
        1. Click Insert Policy. In the Policy Name column, select New Policy.
        2. In the Create Authentication Policy dialog box, specify a name for the policy, select Authentication Type as CERT, and click New.
        3. In the Create Authentication Server dialog box, specify the server Name and the User Name Field and click Create.
        4. Bind the policy to the NetScaler Gateway Vserver with **ns_true** as expression and default priority.
    3. Configure SSL-based client authentication with Client Certificate set as Optional.
7. Continue with the wizard as described in "Configuring a Web Interface Site for Remote Users Using NetScaler Gateway."

To use smart card authentication for a web interface site through NetScaler Gateway by using the command line interface

At the command prompt, type:

add wi site <sitePath> <agURL> <staURL> -authenticationPoint AccessGateway -agAuthenticationMethod SmartCard -defaultAccessMethod GatewayDirect

Additionally, smart card support on web interface requires certificate-based authentication on the NetScaler GatewayAdditionally, smart card support on web interface requires certificate-based authentication on the Access Gateway. If you do not have certificate-based authentication already configured on the NetScaler Gateway vserver, do the following:

1. Configure an NetScaler Gateway vserver and bind the server and CA certificate to it.
   add vpn vserver <vpnvserver_name> SSL <AccessGateway_VIP> <port>

   bind ssl vserver <vpnvserver_name> -certkeyName <Server_Cert_Key_Pair>

   bind ssl vserver <vpnvserver_name> -certkeyName <Root_Cert_as_CKP> -CA

   set ssl vserver <vpnvserver_name> -clientAuth ENABLED -clientCert Optional

   add dns nameserver <dns_server_ip>

2. Configure a certificate-based authentication policy and bind itcertAction_name to the NetScaler Gateway vserver.
   add authentication certAction <certAction_name> -userNameField <string>

   add authentication certPolicy <certPolicy_name> <rule> <certAction_name>

   bind vpn vserver <vserver_name> -policy <certPolicy_name>

3. Configure the web interface site in ICA proxy mode through a session policy and then bind it to NetScaler Gateway vserver.
   add vpn sessionAction <sessionAction_name> -defaultAuthorizationAction ALLOW -SSO ON -wihome "<WI_URL>" -wiPortalMode NORMAL -icaproxy ON

   add vpn sessionPolicy <sessionPolicy_name> <rule> <sessionAction_name>

   bind vpn vserver <vpnvserver_name> -policyName <sessionPolicy_name>

   bind vpn vserver <vpnvserver_name> -staServer "<sta_server_url>"

Using the WebInterface.conf Dialog Box

Aug 08, 2014

The WebInterface.conf dialog box in the configuration utility displays the content of the webinterface.conf file for a Web Interface site.

You can do the following from this dialog box:

- Edit the WebInterface.conf file and save the changes.
- Search the file's content for instances of a text string.
- Easily save the WebInterface.conf file to your local computer.

To search a string in the webinterface.conf file by using the configuration utility

1. Navigate to System > Web Interface > Sites, select the web interface site, and click WebInterface.conf.
2. In the WebInterface.conf dialog box, use the following controls:
   - Find. Displays the following search options that you can use to find one or more instances of a text string in a configuration:
     - Look for. Provides a space for you to type the text string that you want to locate in the configuration. As you type the text, the first instance is displayed. If the word you are looking for is not in the file, the Look for text box will change color.
     - Next. Finds and highlights the next occurrence of the text string you typed in Look for.
     - Previous. Finds and highlights the previous occurrence of the text string you typed in Look for.
     - Mark All. Highlights all instances of the text string at one time you typed in Look for. Scroll to review each highlighted instance.

To save the content of the webinterface.conf to your local system by using the configuration utility

1. Navigate to System > Web Interface > Sites, click WebInterface.conf, and select Save output text to a file.

Using the config.xml Dialog Box

Aug 08, 2014

The Config.xml dialog box in the configuration utility displays the content of the config.xml file for a Web Interface site of site type XenApp/XenDesktop Services Site.

You can do the following from this dialog box:

- Edit the config.xml file and save the changes.
- Search the file's content for instances of a text string.
- Easily save the config.xml file to your local computer.

To search a string in the config.xml file by using the configuration utility

1. Navigate to System > Web Interface > Sites, select XenApp/XenDesktop services site, and click Config.xml.
2. In the Config.xml dialog box, use the following controls:
   - Find. Displays the following search options that you can use to find one or more instances of a text string in a configuration:
     - Look for. Provides a space for you to type the text string that you want to locate in the configuration. As you type the text, the first instance is displayed. If the word you are looking for is not in the file, the Look for text box will change color.
     - Next. Finds and highlights the next occurrence of the text string you typed in Look for.
     - Previous. Finds and highlights the previous occurrence of the text string you typed in Look for.
     - Mark All. Highlights all instances of the text string at one time you typed in Look for. Scroll to review each highlighted instance.

To save the content of the config.xml to the local system by using the configuration utility

1. Navigate to System > Web Interface > Sites, and select XenApp/XenDesktop Services Site.
2. Click Config.xml and select Save output text to a file.

Getting Started with Citrix NetScaler

Mar 15, 2012

Intended for system and network administrators who install and configure complex networking equipment, this section of the library describes initial set-up and basic configuration of the NetScaler, including the following topics.

Understanding the NetScaler

Updated: 2013-10-28

The Citrix NetScaler product is an application switch that performs application-specific traffic analysis to intelligently distribute, optimize, and secure Layer 4-Layer 7 (L4–L7) network traffic for web applications. For example, a NetScaler bases load balancing decisions on individual HTTP requests instead of on long-lived TCP connections, so that the failure or slowdown of a server is managed much more quickly and with less disruption to clients. The NetScaler feature set can be broadly categorized as consisting of switching features, security and protection features, and server-farm optimization features.

Switching Features

Updated: 2013-09-06

When deployed in front of application servers, a NetScaler ensures optimal distribution of traffic by the way in which it directs client requests. Administrators can segment application traffic according to information in the body of an HTTP or TCP request, and on the basis of L4–L7 header information such as URL, application data type, or cookie. Numerous load balancing algorithms and extensive server health checks improve application availability by ensuring that client requests are directed to the appropriate servers.

Security and Protection Features

Updated: 2013-10-28

NetScaler security and protection features protect web applications from Application Layer attacks. A NetScaler allows legitimate client requests and can block malicious requests. It provides built-in defenses against denial-of-service (DoS) attacks and supports features that protect against legitimate surges in application traffic that would otherwise overwhelm the servers. An available built-in firewall protects web applications from Application Layer attacks, including buffer overflow exploits, SQL injection attempts, cross-site scripting attacks, and more. In addition, the firewall provides identity theft protection by securing confidential corporate information and sensitive customer data.

Optimization Features

Updated: 2013-09-06

Optimization features offload resource-intensive operations, such as Secure Sockets Layer (SSL) processing, data compression, client keep-alive, TCP buffering, and the caching of static and dynamic content from servers. This improves the performance of the servers in the server farm and therefore speeds up applications. A NetScaler supports several transparent TCP optimizations, which mitigate problems caused by high latency and congested network links, accelerating the delivery of applications while requiring no configuration changes to clients or servers.

Understanding Policies and Expressions

A policy defines specific details of traffic filtering and management on a NetScaler. It consists of two parts: the expression and the action. The expression defines the types of requests that the policy matches. The action tells the NetScaler what to do when a request matches the expression. As an example, the expression might be to match a specific URL pattern to a type of security attack, with the action being to drop or reset the connection. Each policy has a priority, and the priorities determine the order in which the policies are evaluated.

When a NetScaler receives traffic, the appropriate policy list determines how to process the traffic. Each policy on the list contains one or more expressions, which together define the criteria that a connection must meet to match the policy.

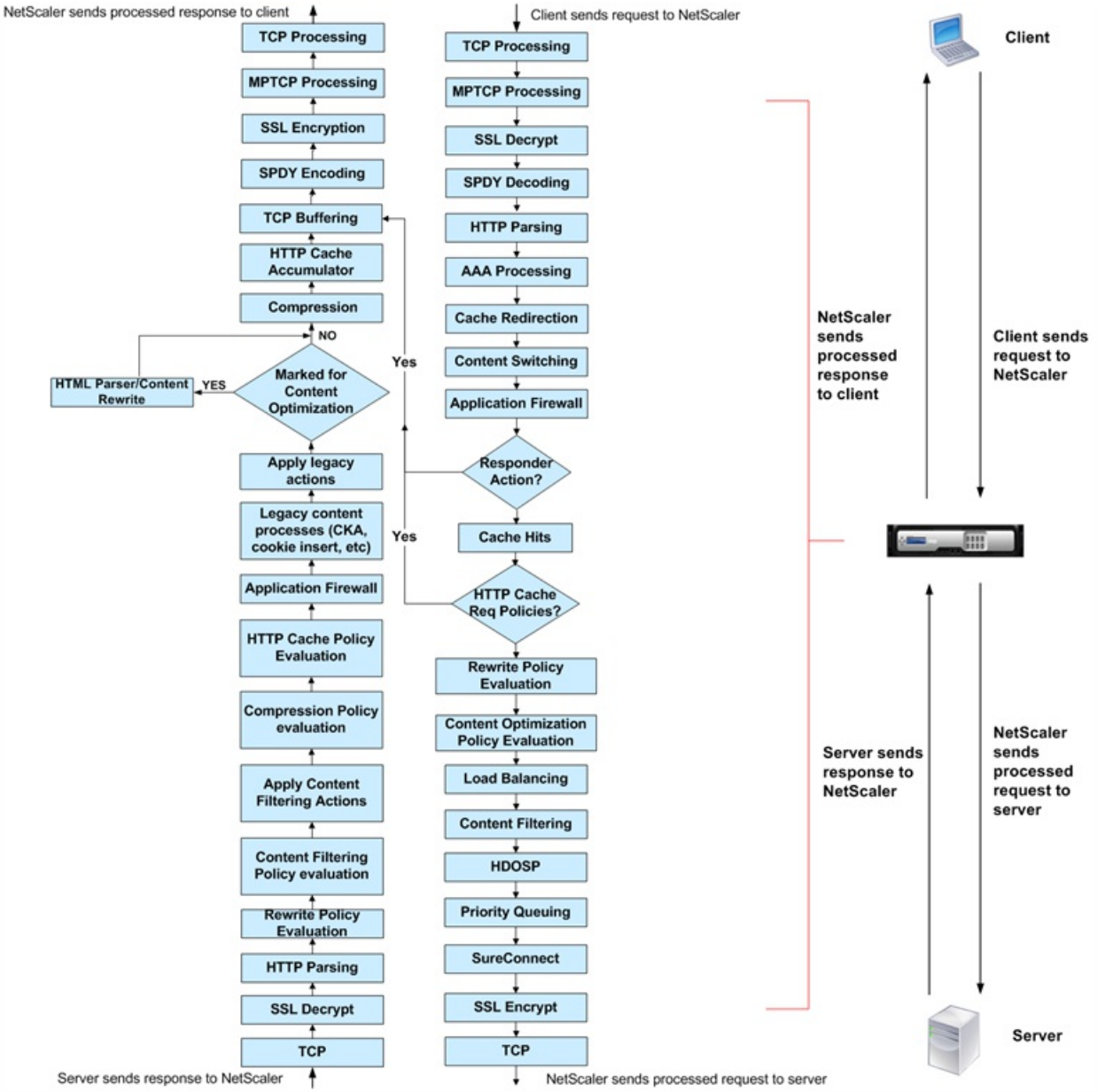
For all policy types except Rewrite policies, a NetScaler implements only the first policy that a request matches, not any additional policies that it might also match. For Rewrite policies, the NetScaler evaluates the policies in order and, in the case of multiple matches, performs the associated actions in that order. Policy priority is important for getting the results you want.

Processing Order of Features

Depending on requirements, you can choose to configure multiple features. For example, you might choose to configure both compression and SSL offload. As a result, an outgoing packet might be compressed and then encrypted before being sent to the client.

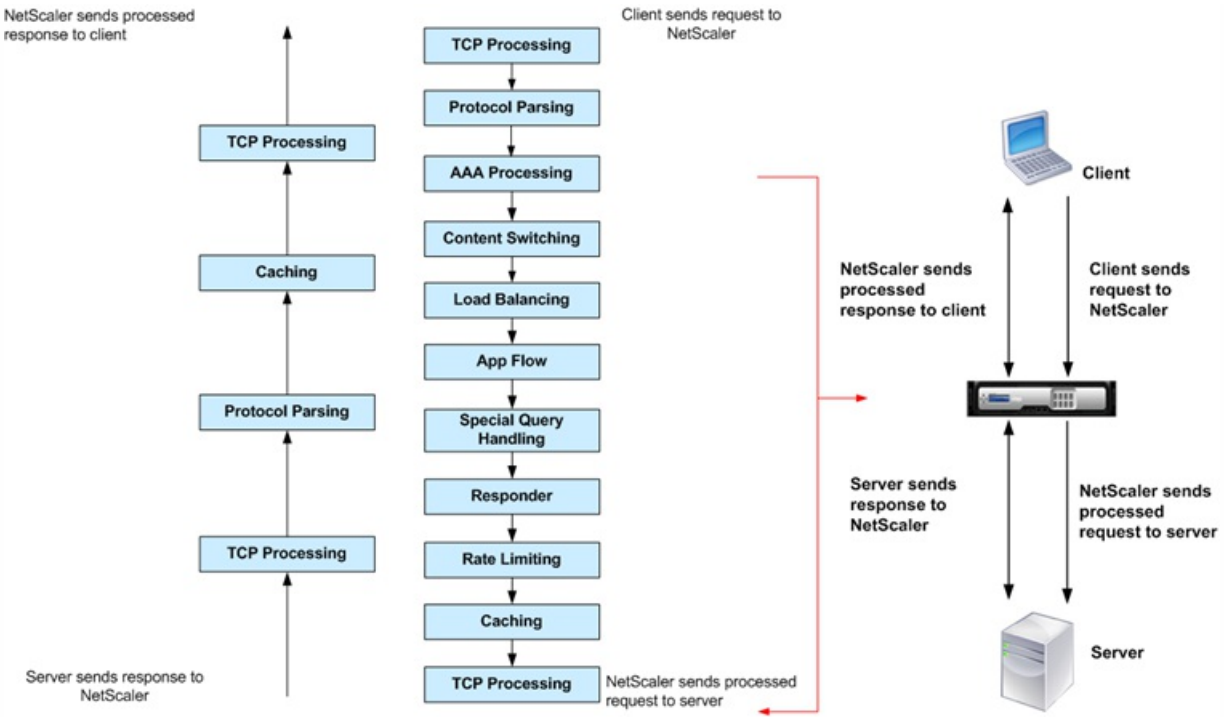
The following figure shows the L7 packet flow in the NetScaler.

Figure 1. L7 Packet Flow Diagram



The following figure shows the DataStream packet flow in the NetScaler. DataStream is supported for MySQL and MS SQL databases. For information about the DataStream feature, see "DataStream."

Figure 2. DataStream Packet Flow Diagram



Where Does a NetScaler Appliance Fit in the Network?

Jun 24, 2013

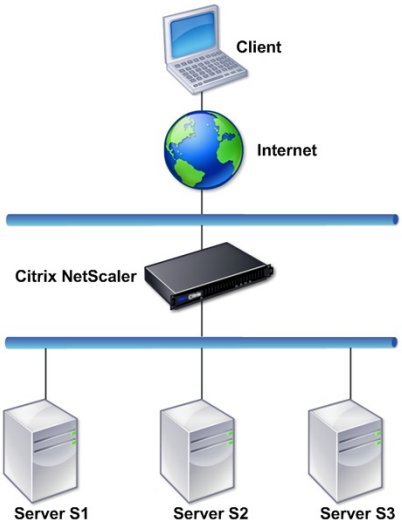
A NetScaler appliance resides between the clients and the servers, so that client requests and server responses pass through it. In a typical installation, virtual servers configured on the appliance provide connection points that clients use to access the applications behind the appliance. In this case, the appliance owns public IP addresses that are associated with its virtual servers, while the real servers are isolated in a private network. It is also possible to operate the appliance in a transparent mode as an L2 bridge or L3 router, or even to combine aspects of these and other modes.

Physical Deployment Modes

Updated: 2013-09-04

A NetScaler appliance logically residing between clients and servers can be deployed in either of two physical modes: inline and one-arm. In inline mode, multiple network interfaces are connected to different Ethernet segments, and the appliance is placed between the clients and the servers. The appliance has a separate network interface to each client network and a separate network interface to each server network. The appliance and the servers can exist on different subnets in this configuration. It is possible for the servers to be in a public network and the clients to directly access the servers through the appliance, with the appliance transparently applying the L4-L7 features. Usually, virtual servers (described later) are configured to provide an abstraction of the real servers. The following figure shows a typical inline deployment.

Figure 1. Inline Deployment



In one-arm mode, only one network interface of the appliance is connected to an Ethernet segment. The appliance in this case does not isolate the client and server sides of the network, but provides access to applications through configured virtual servers. One-arm mode can simplify network changes needed for NetScaler installation in some environments.

For examples of inline (two-arm) and one-arm deployment, see "Understanding Common Network Topologies."

Citrix NetScaler as an L2 Device

Updated: 2013-09-04

A NetScaler functioning as an L2 device is said to operate in L2 mode. In L2 mode, the NetScaler forwards packets between network interfaces when all of the following conditions are met:

- The packets are destined to another device's media access control (MAC) address.
- The destination MAC address is on a different network interface.
- The network interface is a member of the same virtual LAN (VLAN).

By default, all network interfaces are members of a pre-defined VLAN, VLAN 1. Address Resolution Protocol (ARP) requests and responses are forwarded to all network interfaces that are members of the same VLAN. To avoid bridging loops, L2 mode must be disabled if another L2 device is working in parallel with the NetScaler.

For information about how the L2 and L3 modes interact, see "Configuring Modes of Packet Forwarding."

For information about configuring L2 mode, see "Enabling and Disabling Layer 2 Mode."

Citrix NetScaler as a Packet Forwarding Device

Updated: 2014-03-14

A NetScaler appliance can function as a packet forwarding device, and this mode of operation is called L3 mode. With L3 mode enabled, the appliance forwards any received unicast packets that are destined for an IP address that does not belong to the appliance, if there is a route to the destination. The appliance can also route packets between VLANs.

In both modes of operation, L2 and L3, the appliance generally drops packets that are in:

- Multicast frames
- Unknown protocol frames destined for an appliance's MAC address (non-IP and non-ARP)
- Spanning Tree protocol (unless BridgeBPDUs is ON)

For information about how the L2 and L3 modes interact, see "Configuring Modes of Packet Forwarding."

For information about configuring the L3 mode, see "Enabling and Disabling Layer 3 Mode."

How a NetScaler Communicates with Clients and Servers

Aug 30, 2013

A NetScaler appliance is usually deployed in front of a server farm and functions as a transparent TCP proxy between clients and servers, without requiring any client-side configuration. This basic mode of operation is called Request Switching technology and is the core of NetScaler functionality. Request Switching enables an appliance to multiplex and offload the TCP connections, maintain persistent connections, and manage traffic at the request (application layer) level. This is possible because the appliance can separate the HTTP request from the TCP connection on which the request is delivered.

Depending on the configuration, an appliance might process the traffic before forwarding the request to a server. For example, if the client attempts to access a secure application on the server, the appliance might perform the necessary SSL processing before sending traffic to the server.

To facilitate efficient and secure access to server resources, an appliance uses a set of IP addresses collectively known as NetScaler-owned IP addresses. To manage your network traffic, you assign NetScaler-owned IP addresses to virtual entities that become the building blocks of your configuration. For example, to configure load balancing, you create virtual servers to receive client requests and distribute them to services, which are entities representing the applications on your servers.

Understanding NetScaler-Owned IP Addresses

Updated: 2014-03-12

To function as a proxy, a NetScaler appliance uses a variety of IP addresses. The key NetScaler-owned IP addresses are:

NetScaler IP (NSIP) address

The NSIP address is the IP address for management and general system access to the appliance itself, and for communication between appliances in a high availability configuration.

Mapped IP (MIP) address

A MIP address is used for server-side connections. It is not the IP address of the appliance. In most cases, when the appliance receives a packet, it replaces the source IP address with a MIP address before sending the packet to the server. With the servers abstracted from the clients, the appliance manages connections more efficiently.

Virtual server IP (VIP) address

A VIP address is the IP address associated with a virtual server. It is the public IP address to which clients connect. An appliance managing a wide range of traffic may have many VIPs configured.

Subnet IP (SNIP) address

A SNIP address is used in connection management and server monitoring. You can specify multiple SNIP addresses for each subnet. SNIP addresses can be bound to a VLAN.

IP Set

An IP set is a set of IP addresses, which are configured on the appliance as SNIP . An IP set is identified with a meaningful name that helps in identifying the usage of the IP addresses contained in it.

Net Profile

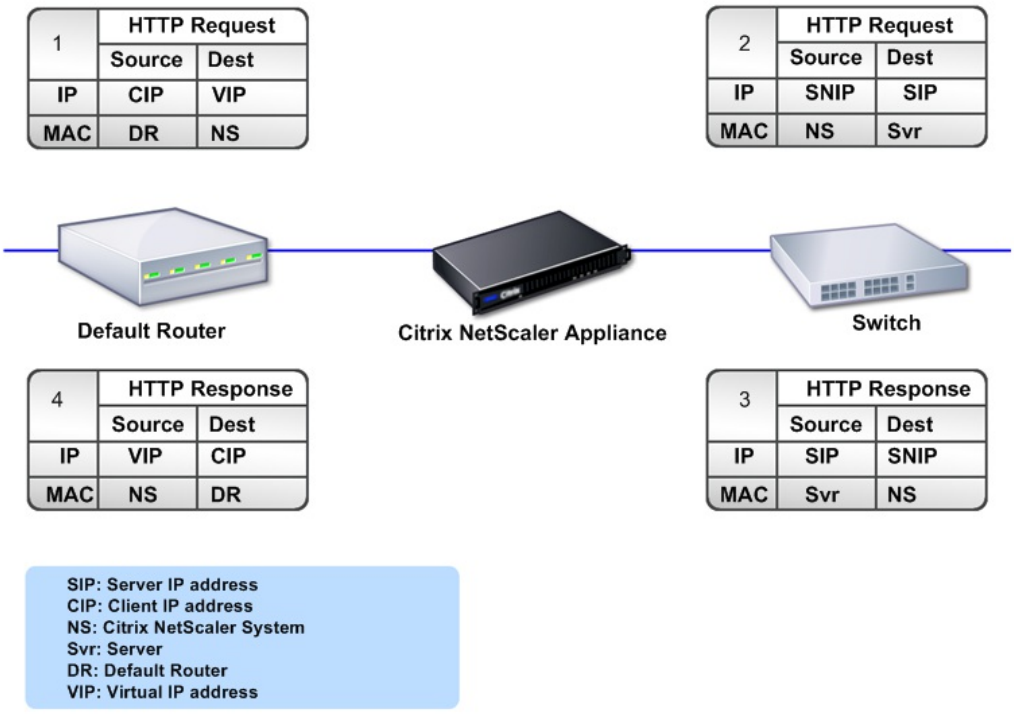
A net profile (or network profile) contains an IP address or an IP set. A net profile can be bound to load balancing or content switching virtual servers, services, service groups, or monitors. During communication with physical servers or peers, the appliance uses the addresses specified in the profile as source IP addresses.

How Traffic Flows Are Managed

Updated: 2014-03-12

Because a NetScaler appliance functions as a TCP proxy, it translates IP addresses before sending packets to a server. When you configure a virtual server, clients connect to a VIP address on the NetScaler instead of directly connecting to a server. As determined by the settings on the virtual server, the appliance selects an appropriate server and sends the client's request to that server. By default, the appliance uses a SNIP address to establish connections with the server, as shown in the following figure.

Figure 1. Virtual Server Based Connections



In the absence of a virtual server, when an appliance receives a request, it transparently forwards the request to the server. This is called the transparent mode of operation. When operating in transparent mode, an appliance translates the source IP addresses of incoming client requests to the SNIP address but does not change the destination IP address. For this mode to work, L2 or L3 mode has to be configured appropriately.

For cases in which the servers need the actual client IP address, the appliance can be configured to modify the HTTP header by inserting the client IP address as an additional field, or configured to use the client IP address instead of a SNIP address for connections to the servers.

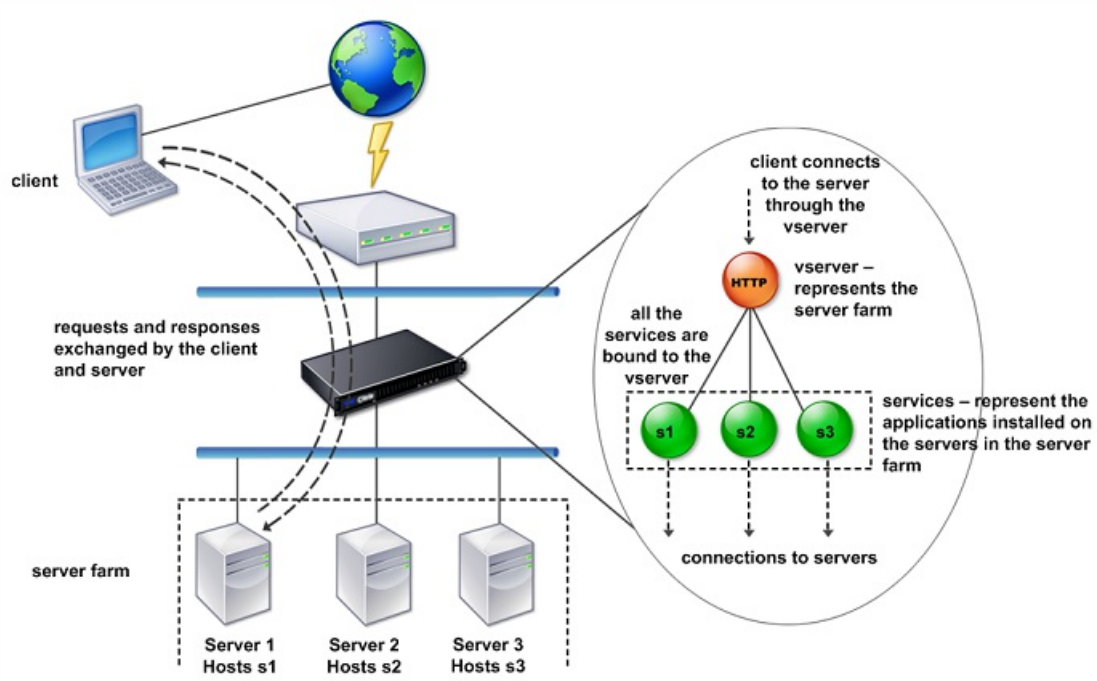
Traffic Management Building Blocks

Updated: 2013-06-24

The configuration of a NetScaler appliance is typically built up with a series of virtual entities that serve as building blocks for traffic management. The building block approach helps separate traffic flows. Virtual entities are abstractions, typically representing IP addresses, ports, and protocol handlers for processing traffic. Clients access applications and resources through these virtual entities. The most commonly used entities are virtual servers and services. Virtual servers represent groups of servers in a server farm or remote network, and services represent specific applications on each server.

Most features and traffic settings are enabled through virtual entities. For example, you can configure an appliance to compress all server responses to a client that is connected to the server farm through a particular virtual server. To configure the appliance for a particular environment, you need to identify the appropriate features and then choose the right mix of virtual entities to deliver them. Most features are delivered through a cascade of virtual entities that are bound to each other. In this case, the virtual entities are like blocks being assembled into the final structure of a delivered application. You can add, remove, modify, bind, enable, and disable the virtual entities to configure the features. The following figure shows the concepts covered in this section.

Figure 2. How Traffic Management Building Blocks Work



A Simple Load Balancing Configuration

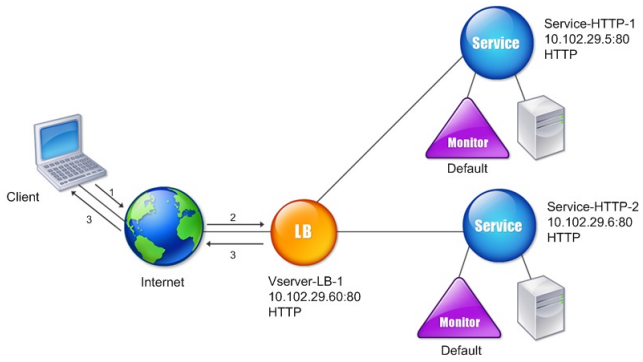
Updated: 2013-08-30

In the example shown in the following figure, the NetScaler appliance is configured to function as a load balancer. For this configuration, you need to configure virtual entities specific to load balancing and bind them in a specific order. As a load balancer, an appliance distributes client requests across several servers and thus optimizes the utilization of resources.

The basic building blocks of a typical load balancing configuration are services and load balancing virtual servers. The services represent the applications on the servers. The virtual servers abstract the servers by providing a single IP address to which the clients connect. To ensure that client requests are sent to a server, you need to bind each service to a virtual server. That is, you must create services for every server and bind the services to a virtual server. Clients use the VIP address to

connect to a NetScaler appliance. When the appliance receives client requests sent to the VIP address, it sends them to a server determined by the load balancing algorithm. Load balancing uses a virtual entity called a monitor to track whether a specific configured service (server plus application) is available to receive requests.

Figure 3. Load Balancing Virtual Server, Services, and Monitors



In addition to configuring the load balancing algorithm, you can configure several parameters that affect the behavior and performance of the load balancing configuration. For example, you can configure the virtual server to maintain persistence based on source IP address. The appliance then directs all requests from any specific IP address to the same server.

Understanding Virtual Servers

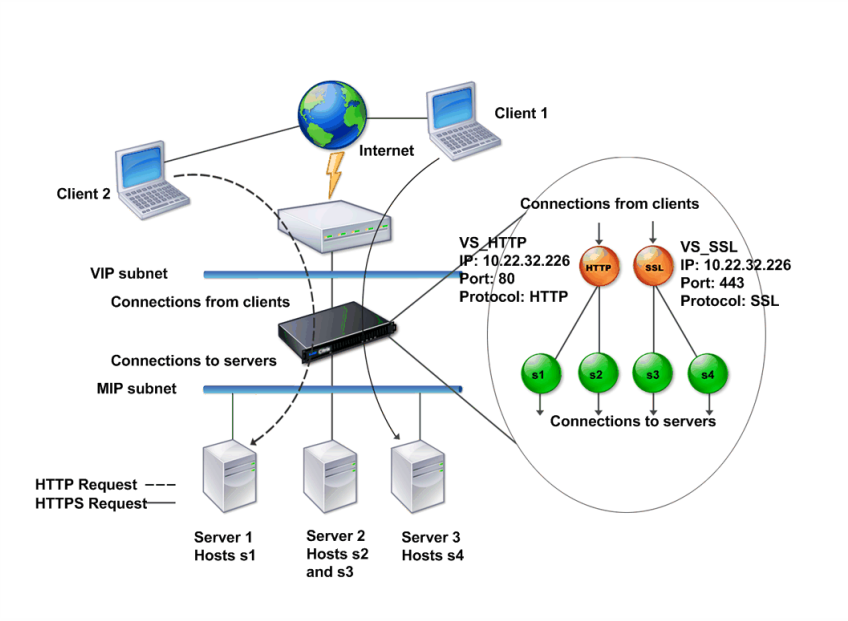
Updated: 2013-09-06

A virtual server is a named NetScaler entity that external clients can use to access applications hosted on the servers. It is represented by an alphanumeric name, virtual IP (VIP) address, port, and protocol. The name of the virtual server is of only local significance and is designed to make the virtual server easier to identify. When a client attempts to access applications on a server, it sends a request to the VIP instead of the IP address of the physical server. When the appliance receives a request at the VIP address, it terminates the connection at the virtual server and uses its own connection with the server on behalf of the client. The port and protocol settings of the virtual server determine the applications that the virtual server represents. For example, a web server can be represented by a virtual server and a service whose port and protocol are set to 80 and HTTP, respectively. Multiple virtual servers can use the same VIP address but different protocols and ports.

Virtual servers are points for delivering features. Most features, like compression, caching, and SSL offload, are normally enabled on a virtual server. When the appliance receives a request at a VIP address, it chooses the appropriate virtual server by the port on which the request was received and its protocol. The appliance then processes the request as appropriate for the features configured on the virtual server.

In most cases, virtual servers work in tandem with services. You can bind multiple services to a virtual server. These services represent the applications running on physical servers in a server farm. After the appliance processes requests received at a VIP address, it forwards them to the servers as determined by the load balancing algorithm configured on the virtual server. The following figure illustrates these concepts.

Figure 4. Multiple Virtual Servers with a Single VIP Address



The preceding figure shows a configuration consisting of two virtual servers with a common VIP address but different ports and protocols. Each of the virtual servers has two services bound to it. The services s1 and s2 are bound to VS_HTTP and represent the HTTP applications on Server 1 and Server 2. The services s3 and s4 are bound to VS_SSL and represent the SSL applications on Server 2 and Server 3 (Server 2 provides both HTTP and SSL applications). When the appliance receives an HTTP request at the VIP address, it processes the request as specified by the settings of VS_HTTP and sends it to either Server 1 or Server 2. Similarly, when the appliance receives an HTTPS request at the VIP address, it processes it as specified by the settings of VS_SSL and it sends it to either Server 2 or Server 3.

Virtual servers are not always represented by specific IP addresses, port numbers, or protocols. They can be represented by wildcards, in which case they are known as wildcard virtual servers. For example, when you configure a virtual server with a wildcard instead of a VIP, but with a specific port number, the appliance intercepts and processes all traffic conforming to that protocol and destined for the predefined port. For virtual servers with wildcards instead of VIPs and port numbers, the appliance intercepts and processes all traffic conforming to the protocol.

Virtual servers can be grouped into the following categories:

Load balancing virtual server

Receives and redirects requests to an appropriate server. Choice of the appropriate server is based on which of the various load balancing methods the user configures.

Cache redirection virtual server

Redirects client requests for dynamic content to origin servers, and requests for static content to cache servers. Cache redirection virtual servers often work in conjunction with load balancing virtual servers.

Content switching virtual server

Directs traffic to a server on the basis of the content that the client has requested. For example, you can create a content switching virtual server that directs all client requests for images to a server that serves images only. Content switching virtual servers often work in conjunction with load balancing virtual servers.

Virtual private network (VPN) virtual server

Decrypts tunneled traffic and sends it to intranet applications.

SSL virtual server

Receives and decrypts SSL traffic, and then redirects to an appropriate server. Choosing the appropriate server is similar to choosing a load balancing virtual server.

Understanding Services

Updated: 2014-03-12

Services represent applications on a server. While services are normally combined with virtual servers, in the absence of a virtual server, a service can still manage application-specific traffic. For example, you can create an HTTP service on a NetScaler appliance to represent a web server application. When the client attempts to access a web site hosted on the web server, the appliance intercepts the HTTP requests and creates a transparent connection with the web server.

In service-only mode, an appliance functions as a proxy. It terminates client connections, uses a SNIP address to establish a connection to the server, and translates the destination IP addresses of incoming client requests to a SNIP address. Although the clients send requests directly to the IP address of the server, the server sees them as coming from the SNIP address. The appliance translates the IP addresses, port numbers, and sequence numbers.

A service is also a point for applying features. Consider the example of SSL acceleration. To use this feature, you must create an SSL service and bind an SSL certificate to the service. When the appliance receives an HTTPS request, it decrypts the traffic and sends it, in clear text, to the server. Only a limited set of features can be configured in the service-only case.

Services use entities called monitors to track the health of applications. Every service has a default monitor, which is based on the service type, bound to it. As specified by the settings configured on the monitor, the appliance sends probes to the application at regular intervals to determine its state. If the probes fail, the appliance marks the service as down. In such cases, the appliance responds to client requests with an appropriate error message or re-routes the request as determined by the configured load balancing policies.

Introduction to the Citrix NetScaler Product Line

Sep 04, 2013

The Citrix NetScaler product line optimizes delivery of applications over the Internet and private networks, combining application-level security, optimization, and traffic management into a single, integrated appliance. You install a NetScaler appliance in your server room and route all connections to your managed servers through it. The NetScaler features that you enable and the policies you set are then applied to incoming and outgoing traffic.

A NetScaler can be integrated into any network as a complement to existing load balancers, servers, caches, and firewalls. It requires no additional client or server side software, and can be configured using the NetScaler web-based GUI and CLI configuration utilities.

NetScaler appliances are available in a variety of hardware platforms that have a range of specifications, including multicore processors.

The NetScaler operating system is the base operating system for all NetScaler hardware platforms. The NetScaler operating system is available in three editions: Standard, Enterprise, and Platinum.

Citrix NetScaler Hardware Platforms

Updated: 2013-11-08

NetScaler hardware is available in a variety of platforms that have a range of hardware specifications, including multicore processors. All hardware platforms support some combination of Fast Ethernet, Gigabit Ethernet, and 10 Gigabit Ethernet interfaces.

The following platforms are available for NetScaler .

- Citrix NetScaler MPX 5500
- Citrix NetScaler MPX 5550/5650
- Citrix NetScaler MPX 7500/9500
- Citrix NetScaler MPX 8200/8400/8600
- Citrix NetScaler MPX 9700/10500/12500/15500
- Citrix NetScaler MPX 11500/13500/14500/16500/18500/20500
- Citrix NetScaler MPX 15000
- Citrix NetScaler MPX 17000
- Citrix NetScaler MPX 17500/19500/21500
- Citrix NetScaler MPX 17550/19550/20550/21550
- Citrix NetScaler MPX 22040/22060/22080/22100/22120

For more information about the hardware platform specifications, see "Introduction to the Hardware Platforms."

The following tables list different editions of the NetScaler and the hardware platforms on which they are available.

Table 1. Product Editions and MPX Hardware Platforms

Hardware	MPX 5500	MPX 5550/5650	MPX 7500/9500	MPX 8200/8400/8600	MPX 15000	MPX 17000
Platinum Edition	Yes	Yes	Yes	Yes	Yes	Yes
Enterprise Edition	Yes	Yes	Yes	Yes	Yes	Yes
Standard Edition	Yes	Yes	Yes	Yes	Yes	Yes

Table 2. Product Editions and MPX Hardware Platforms (contd.)

Hardware	MPX 9700/10500/12500/15500	MPX 11500/13500/14500/16500/18500/20500	MPX 17500/19500/21500	MPX 17550/19550/20550/21550	MPX 22040/22060/22080/22100/22120
Platinum Edition	Yes	Yes	Yes	Yes	Yes
Enterprise Edition	Yes	Yes	Yes	Yes	Yes
Standard Edition	Yes	Yes	Yes	Yes	Yes

Citrix NetScaler Editions

Updated: 2013-09-04

The NetScaler operating system is available in Standard, Enterprise, and Platinum editions. The Enterprise and Standard editions have limited features available. Feature licenses are required for all editions.

For instructions on how to obtain and install licenses, see "http://support.citrix.com/article/ctx121062."

The Citrix NetScaler editions are described as follows:

- *— Citrix NetScaler, Standard Edition*
  . Provides small and medium enterprises with comprehensive Layer 4- Layer 7 (L4-L7) traffic management, enabling increased web application availability.
- *— Citrix NetScaler, Enterprise Edition*
  . Provides web application acceleration and advanced L4-L7 traffic management, enabling enterprises to increase web application performance and availability and reduce datacenter costs.
- *— Citrix NetScaler, Platinum Edition*
  . Provides a web application delivery solution that reduces data center costs and accelerates application performance, with end-to-end visibility of application performance, and provides advanced application security.

The following table summarizes the features supported by each edition in the Citrix NetScaler product line:

Table 3. Citrix NetScaler Application Delivery Product Line Features

Key Features	Platinum Edition	Enterprise Edition	Standard Edition
Application availability			
Layer 4 load balancing	Yes	Yes	Yes
Layer 7 content switching	Yes	Yes	Yes
AppExpert rate controls	Yes	Yes	Yes
IPv6 support	Yes	Yes	Yes
Global server load balancing (GSLB)	Yes	Yes	Optional
Dynamic routing protocols	Yes	Yes	No
Surge protection	Yes	Yes	No
Priority queuing	Yes	Yes	No
Application acceleration			
Client and server TCP optimizations	Yes	Yes	Yes
Citrix AppCompress for HTTP	Yes	Yes	Optional
Citrix AppCache	Yes	Optional	No
Citrix Branch Repeater client	Yes	No	No
Application security			
Layer 4 DoS defenses	Yes	Yes	Yes
Layer 7 content filtering	Yes	Yes	Yes
HTTP/URL Rewrite	Yes	Yes	Yes
NetScaler Gateway, EE SSL VPN	Yes	Yes	Yes
Layer 7 DoS Defenses	Yes	Yes	No
AAA security	Yes	Yes	No
Application firewall with XML security	Yes	Optional	No
Simple manageability			
AppExpert visual policy builder	Yes	Yes	Yes

Key Features	Platinum Edition	Enterprise Edition	Standard Edition
AppExpert service callouts	Yes	Yes	Yes
AppExpert templates	Yes	Yes	Yes
Role-based administration	Yes	Yes	Yes
Configuration wizards	Yes	Yes	Yes
Citrix Command Center	Yes	Yes	No
Citrix EdgeSight for NetScaler	Yes	Optional	No
Web 2.0 optimization			
Rich Internet application support	Yes	Yes	Yes
Advanced server offload	Yes	Yes	No
Lower total cost of ownership (TCO)			
TCP buffering	Yes	Yes	Yes
TCP multiplexing	Yes	Yes	Yes
SSL offload and acceleration	Yes	Yes	Yes
Cache redirection	Yes	Yes	No
Citrix EasyCall	Yes	No	No

Note: While we have taken care to ensure absolute accuracy when compiling this information, it might change. For the latest information, see Citrix Support at "http://www.citrix.com."

Supported Releases on NetScaler Hardware

Updated: 2014-06-30

The following table lists the earliest NetScaler builds for releases that are supported on the NetScaler MPX platforms.

Hardware	Software Release	Software Build #
MPX 5500	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 5550/5650	10.5	All
	10.1	All
	10.0	71.6.nc and later
	9.3	59.5.nc and later
MPX 7500/9500	10.5	All
	10.1	All
	10.0	All

Hardware	Software Release	Software Build #
	9.3	All
MPX 8005/8015	10.5	All
	10.1	122.17.nc and later
	9.3	65.8.nc and later
MPX 8200/8400/8600	10.5	All
	10.1	All
	10.0	70.7.nc and later
	9.3	58.5.nc and later
MPX 9700/10500/12500	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 9700/10500/12500 10G	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 15500	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 15500 10G	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 11500/13500/14500/16500/18500/20500	10.5	All
	10.1	All
	10.0	All
	9.3	52.3.nc and later
MPX 11515/11520/11530/11540/11542	10.5	All
	10.1	123.11.nc and later

Hardware	Software Release	Software Build #
MPX 15000	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 17000	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 17500/19500/21500	10.5	All
	10.1	All
	10.0	All
	9.3	All
MPX 17550/19550/20550/21550	10.5	All
	10.1	All
	10.0	All
	9.3	53.5.nc and later
MPX 22040/22060/22080/22100/22120	10.5	51.10.nc and later
	10.1	123.11.nc and later
	9.3	65.8.nc and later
MPX 24100/24150	10.5	51.10.nc and later
	10.1	129.11.nc and later

Supported Browsers

Updated: 2014-06-24

To access the configuration utility and Dashboard, your workstation must have a supported web browser and version 1.6 or above of the Java applet plug-in installed.

Operating System	Browser	Versions
Windows 7	Internet Explorer	8, 9, and 10
	Mozilla Firefox	3.6.25 and above
	Google Chrome	15 and above
Windows 64 bit	Internet Explorer	8 and 9
	Google Chrome	15 and above

Operating System	Browser	Versions
Mac OS	Mozilla Firefox	12 and above
	Safari	5.1.3
	Google Chrome	15 and above

Software Requirements

Updated: 2013-07-11

Wireshark Versions for NetScaler Releases

The following table provides the Wireshark versions that you can use with different NetScaler releases and builds.

NetScaler Release	Build Number	Wireshark Version
10.1	Later than 95.2	1.9.1
10	Later than 69.3	1.8.4
9.3.e	Later than 57.5004.e	1.9.1
9.3.e	Up to 57.5004.e	1.8.4
9.3	Up to 60.3	1.8.4
9.2	Up to 57.2	1.8.4

Installing the NetScaler Hardware

Sep 04, 2013

Before installing a NetScaler appliance, review the pre-installation checklist. A NetScaler is typically mounted in a rack, and all models ship with rack-rail hardware. All models except the 7000 support small form factor pluggable SFP, XFP, or SFP+ transceivers. After mounting the appliance and installing the transceivers, connect the NetScaler to your network. Use a console cable to connect the NetScaler to a personal computer so that you can perform an initial configuration. After connecting everything else, connect the NetScaler to a power source.

This document includes the following:

Unpacking the Appliance

May 08, 2014

The hardware accessories for your particular appliance, such as cables, adapters, and rail kit, vary depending on the hardware platform you ordered. Unpack the box that contains your new appliance on a sturdy table with plenty of space and inspect the contents.

Use the following list to verify that you received everything that should have been included in the box.

- The appliance you ordered
- One RJ-45 to DB-9 adapter
- One 6 ft RJ-45/DB-9 cable
- The following list specifies the number of power cables included for each appliance model:
  - One power cable for the MPX 5500, MPX 5550/5650, MPX 7500/9500, and MPX 8005/8015/8200/8400/8600/8800 appliances
  - Two power cables for the MPX 15000, MPX 17000, MPX 9700/10500/12500/15500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14000, MPX 17500/19500/21500, and MPX 25100T/25160T appliances
  - Four power cables for the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 appliances
    Note: Make sure that a power outlet is available for each cable.

  Note: For Brazilian customers, Citrix does not ship a power cable. Use a cable that conforms to the **ABNT NBR 14136:2002** standard.
- One standard 4-post rail kit
  Note: If the kit that you received does not fit your rack, contact your Citrix sales representative to order the appropriate kit.

In addition to the items included in the box with your new appliance, you will need the following items to complete the installation and initial configuration process.

- Ethernet cables for each additional Ethernet port that you will connect to your network
- One available Ethernet port on your network switch or hub for each NetScaler Ethernet port you want to connect to your network
  Note: Transceiver modules are sold separately. Contact your Citrix sales representative to order transceiver modules for your appliance. Only transceivers supplied by Citrix are supported on the appliance.
- A computer to serve as a management workstation

Rack Mounting the Appliance

May 08, 2014

Most appliances can be installed in standard server racks that conform to EIA-310-D specification. The appliances ship with a set of rails, which you must install before you mount the appliance. The only tools that you need for installing an appliance are a Phillips screwdriver and a flathead screwdriver.

Caution: If you are installing the appliance as the only unit in the rack, mount it at the bottom. If the rack contains other units, make sure that the heaviest unit is at the bottom. If the rack has stabilizing devices available, install them before mounting the appliance.

The following table lists the different hardware platforms and the rack units required for each platform.

Table 1. ***Height Requirements For Each Platform***

Platform	**Number of rack units**
MPX 5500	One rack unit
MPX 5550/5650	One rack unit
MPX 7500/9500	One rack unit
MPX 8005/8015/8200/8400/8600/8800	One rack unit
MPX 9700/10500/12500/15500	Two rack units
MPX 14000	One rack unit
MPX 15000, MPX 17000	Two rack units
MPX 11500/13500/14500/16500/18500/20500	Two rack units
MPX 11515/11520/11530/11540/11542	Two rack units
MPX 17500/19500/21500	Two rack units
MPX 17550/19550/20550/21550	Two rack units
MPX 22040/22060/22080/22100/22120	Two rack units
MPX 24100/24150	Two rack units

MPX 25100T/25160T	Two rack units

Each appliance ships with a mounting rail kit that contains two rail assemblies, one for the left side and the other for the right side of the appliance, and screws to attach the rails. An assembly consists of an inner rail and a rack rail. The supplied rail kit is 28 inches long (38 inches extended). Contact your Citrix sales representative to order a 23-inch (33 inches extended) rail kit.
Note: The same rail kit is used for both square-hole and round-hole racks. See "Installing the Rail Assembly to the Rack" for specific instructions for threaded, round-hole racks.
To mount the appliance, you must first install the rails and then install the appliance in the rack.

Perform the following tasks to mount the appliance:
- Remove the inner rails from the rail assembly.
- Attach the inner rails to the appliance.
- Install the rack rails on the rack.
- Install the appliance in the rack.

The appliance is shipped with rack-rail hardware. This hardware consists of two inner rails that you attach to the appliance, one on each side, and a rack-rail assembly that you attach to the rack. The following figure illustrates the steps involved in mounting the Citrix NetScaler appliance to a rack.

To remove the inner rails from the rail assembly

1. Place the rail assembly on a flat surface.
2. Slide out the inner rail toward the front of the assembly.
3. Depress the latch until the inner rail comes all the way out of the rail assembly.
4. Repeat steps 1 through 3 to remove the second inner rail.

To attach the inner rails to the appliance

1. Position the right inner rail behind the handle on the right side of the appliance.
2. Align the holes on the rail with the corresponding holes on the side of the appliance.
3. Attach the rail to the appliance with the provided screws: 4 per side for a 1U appliance and 5 per side for a 2U appliance, as shown in the following figure.
   Figure 1. Attaching inner rails

4. Repeat steps 1 through 3 to install the left inner rail on the other side of the appliance.

To install the rack rails on the rack

1. If you have a round-hole, threaded rack, skip to step 3.
2. Install square nut retainers into the front post and back post of the rack as shown in the following figures. Before

inserting a screw, be sure to align the square nut with the correct hole for your 1U or 2U appliance. The three holes are not evenly spaced.

Figure 2. Installing Retainers into the Front Rack Posts

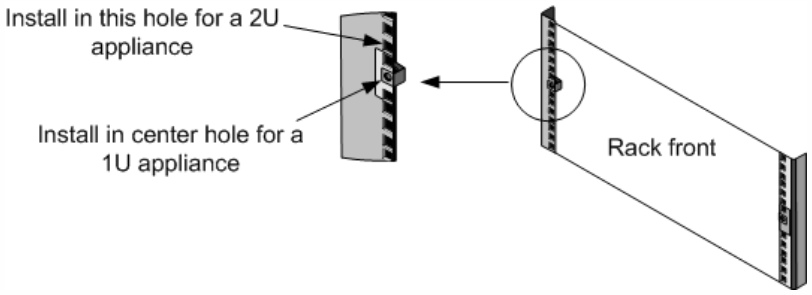
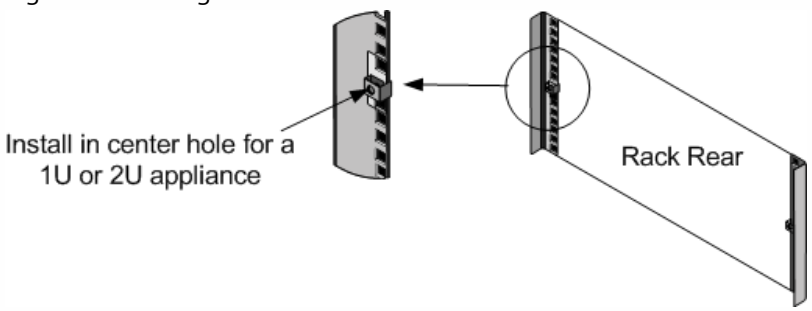
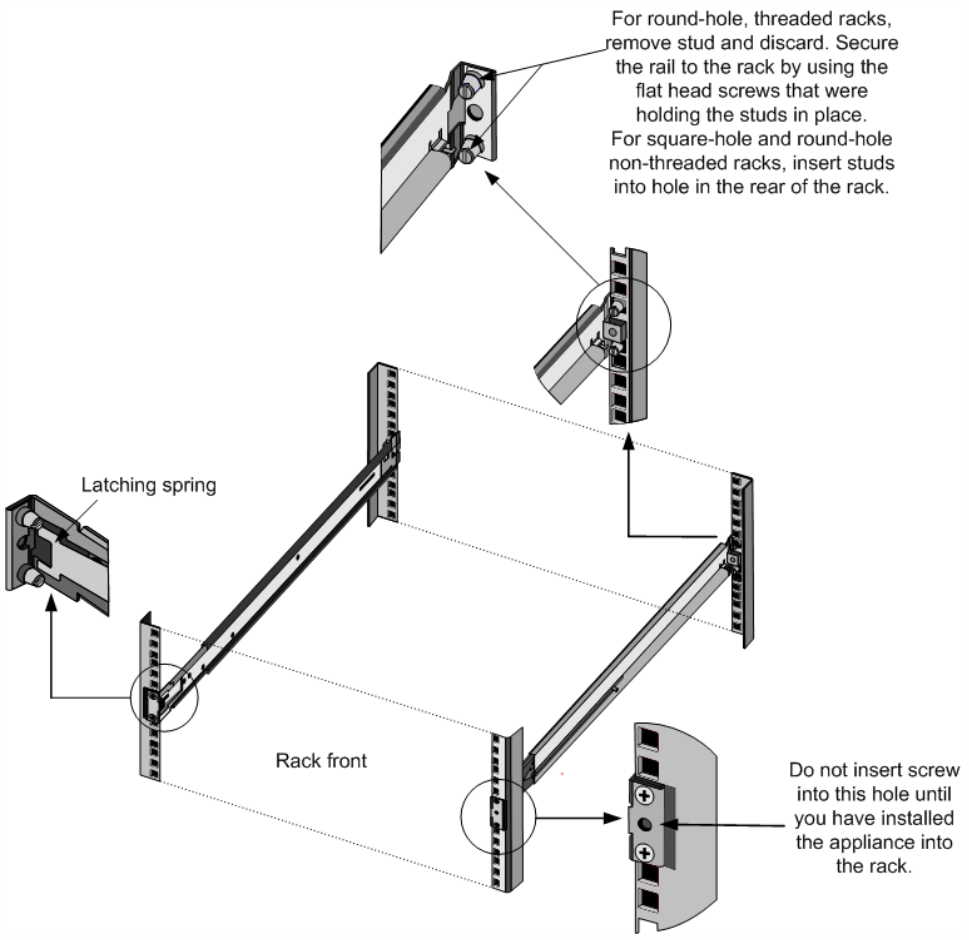


Figure 3. Installing Retainers into the Rear Rack Posts



3. Install the adjustable rail assembly into the rack as shown in the following figures. Use a screw to lock the rear rail flange into the rack. With the screw securing the rail in place, you can optionally remove the latching spring.

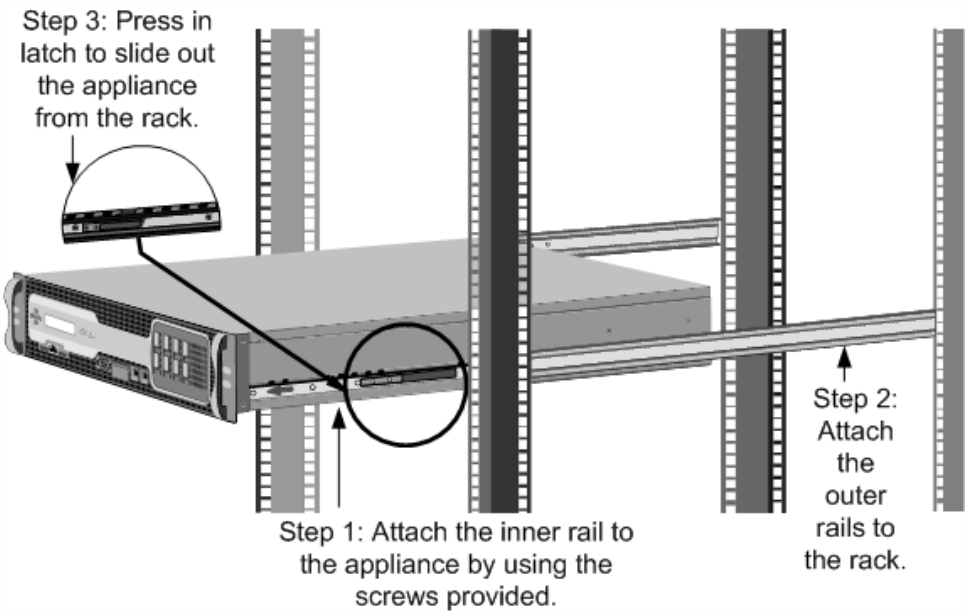
Figure 4. Installing the Rail Assembly to the Rack



To install the appliance in the rack

1. Align the inner rails, attached to the appliance, with the rack rails.
2. Slide the appliance into the rack rails, keeping the pressure even on both sides.
3. Verify that the appliance is locked in place by pulling it all the way out from the rack.

Figure 5. Rack Mounting the Appliance



Installing and Removing 1G SFP Transceivers

Mar 01, 2014

Note: This section applies to the MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 22040/22060/22080/22100/22120, and MPX 24100/24150 appliances.

A Small Form-Factor Pluggable (SFP) is a compact transceiver that can operate at speeds of up to 1 gigabit per second and is available in both copper and fiber types. Inserting a 1G SFP copper transceiver converts the 1G SFP port to a 1000BASE-T port. Inserting a 1G SFP fiber transceiver converts the 1G SFP port to a 1000BASE-X port. Auto-negotiation is enabled by default on the 1G SFP port into which you insert your 1G SFP transceiver. As soon as a link between the port and the network is established, the speed and mode are matched on both ends of the cable.

Note: The 1G SFP transceiver is hot-swappable from release 9.3 build 47.5 and later on the NetScaler appliances that use the e1k interface. The following platforms support 1G SPF transceivers:

- MPX 7500/9500
- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500
- MPX 11500/13500/14500/16500/18500/20500
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150

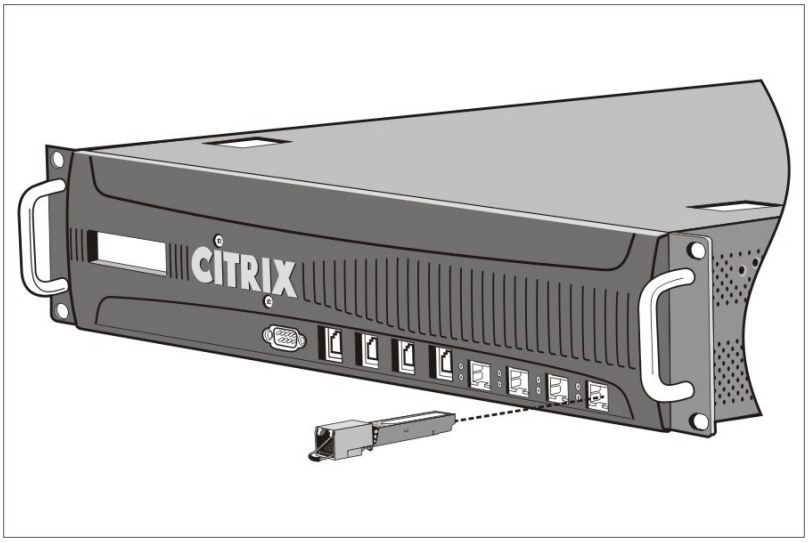
Caution: NetScaler appliances do not support 1G SFP transceivers from vendors other than Citrix Systems. Attempting to install third-party 1G SFP transceivers on your NetScaler appliance voids the warranty.

Insert 1G SFP transceivers into the 1G SFP ports on the front panel of the appliance. Frequent installation and removal of transceivers shortens their life span. Follow the removal procedure carefully to avoid damaging the 1G SFP transceiver or the appliance.

Caution: Do not install the transceivers with the cables attached. Doing so can damage the cable, the connector, or the optical interface of the transceiver.

To install a 1G SFP transceiver

1. Remove the 1G SFP transceiver carefully from its box.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Align the 1G SFP transceiver to the front of the 1G SFP transceiver port on the front panel of the appliance, as shown in the following figure.
   Note: The illustration in the following figures might not represent your actual appliance.
   Figure 1. Installing a 1G SFP transceiver



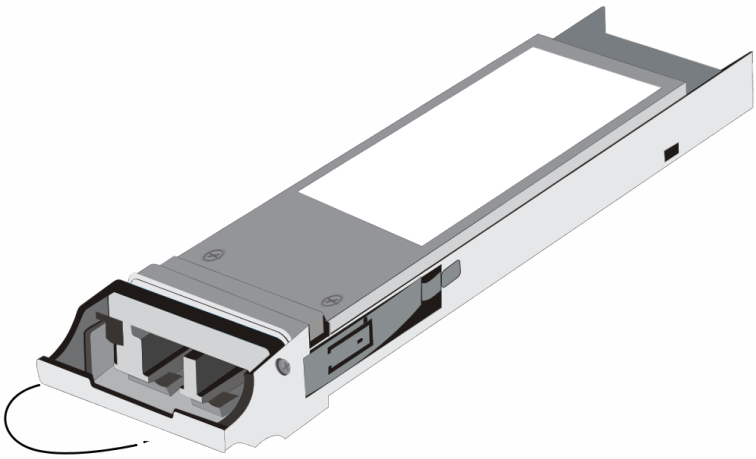
3. Hold the 1G SFP transceiver between your thumb and index finger and insert it into the 1G SFP transceiver port, pressing it in until you hear the transceiver snap into place.
4. Lock the transceiver.
5. Verify that the LED is green and blinks twice, which indicates that the transceiver is functioning correctly.
6. If you are using a fiber 1G SFP transceiver, do not remove the dust caps attached to the transceiver and the cable until you are ready to insert the cable.

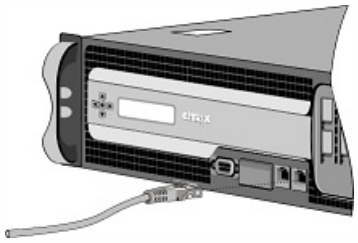
To remove a 1G SFP transceiver

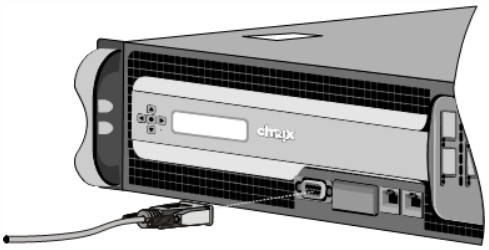
1. Disconnect the cable from the 1G SFP transceiver. If you are using a fiber optic cable, replace the dust cap on the cable before putting it away.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Unlock the 1G SFP transceiver.
3. Hold the 1G SFP transceiver between your thumb and index finger and slowly pull it out of the port.
4. If you are removing a fiber 1G SFP transceiver, replace the dust cap before putting it away.
5. Put the 1G SFP transceiver into its original box or another appropriate container.

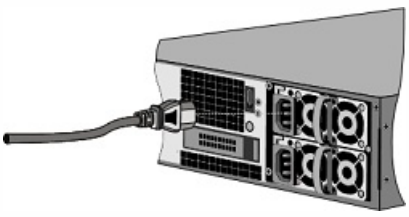
-
-
-
-
-
-
-
-
-
-

-
-









-
-
-

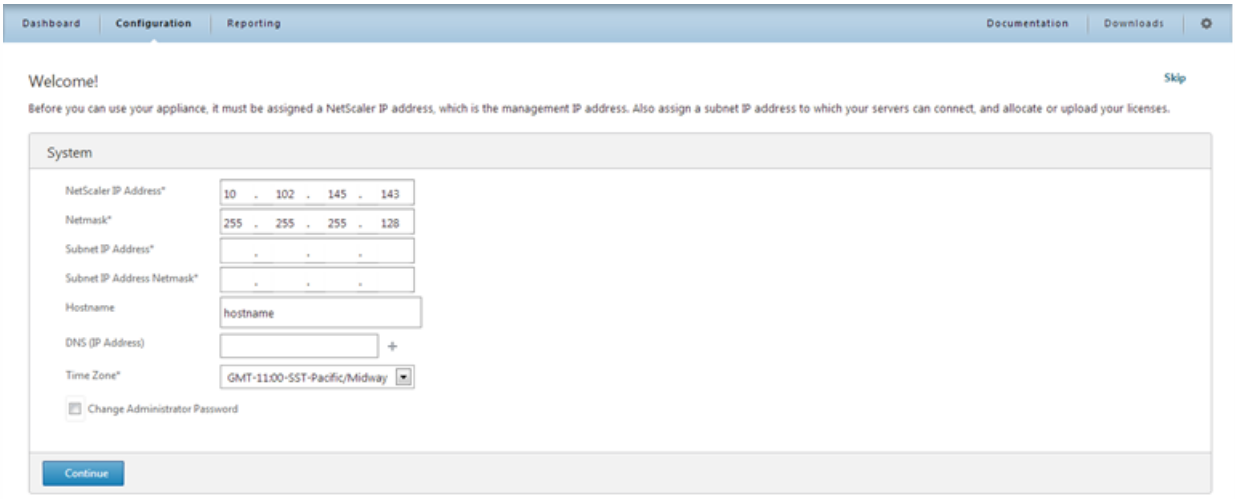
-
-

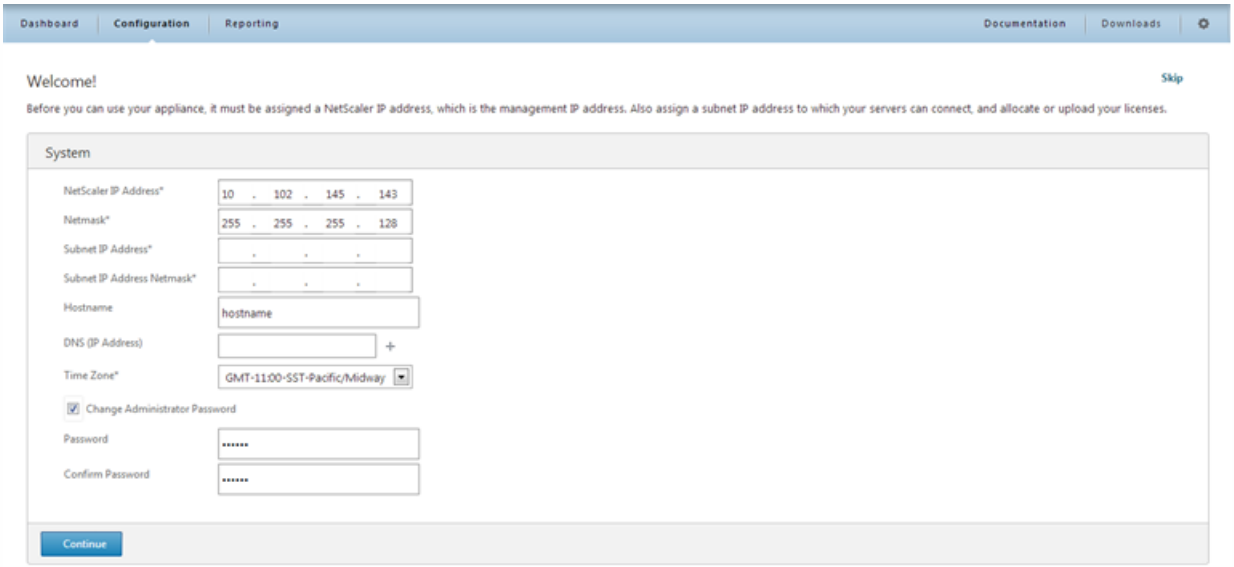
```
login as: nsroot
Using keyboard-interactive authentication.
Password:
Last login: Tue Jun 16 10:37:28 2009 from 10.102.29.9

 Done
>
```

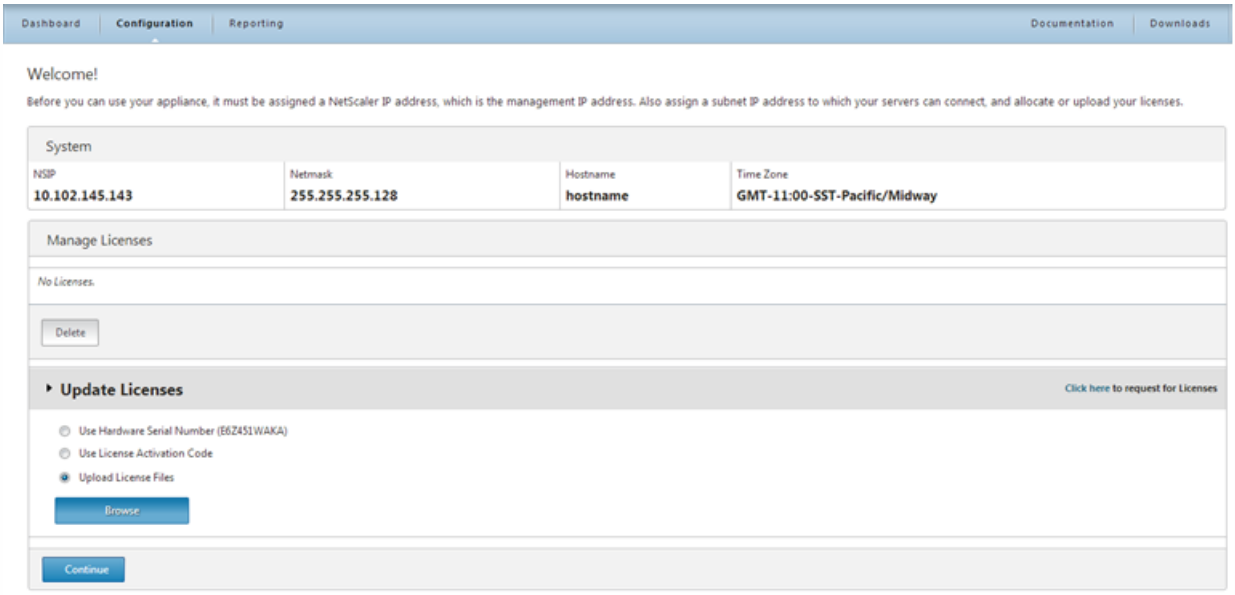
-
-
-

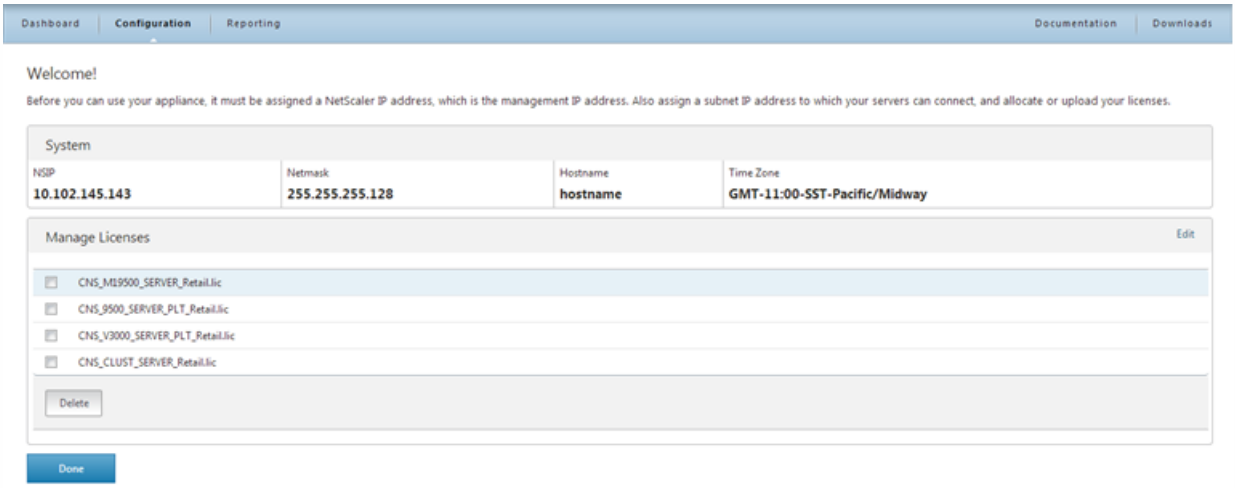
-
-
-
-

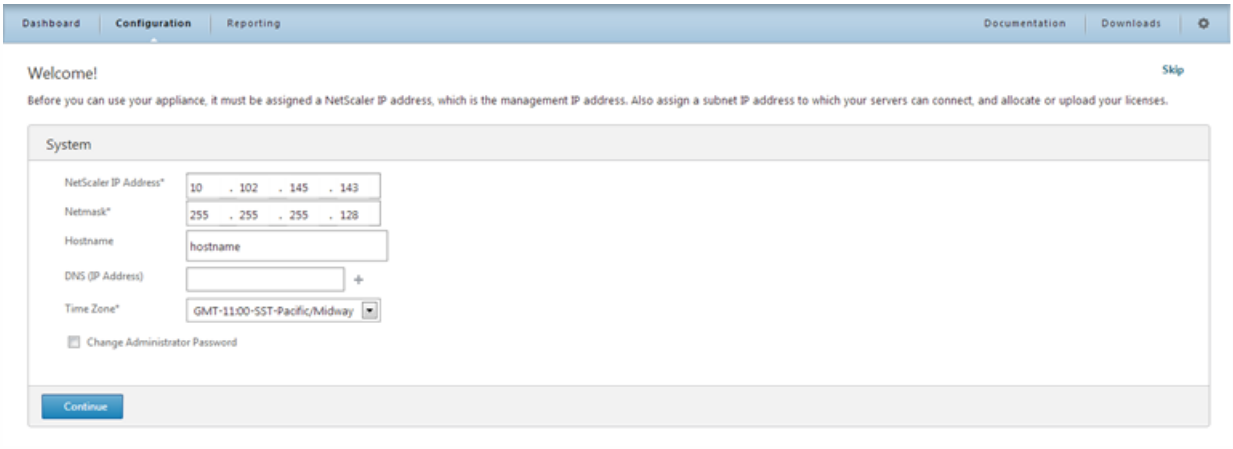
-
-
-
-
-
-
-

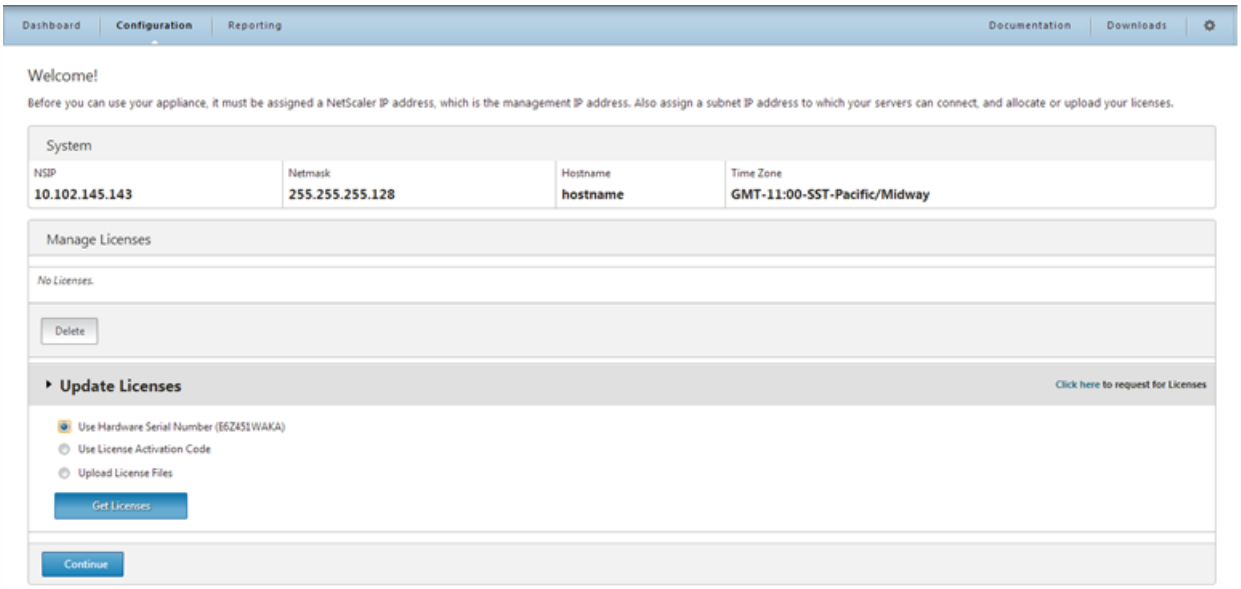


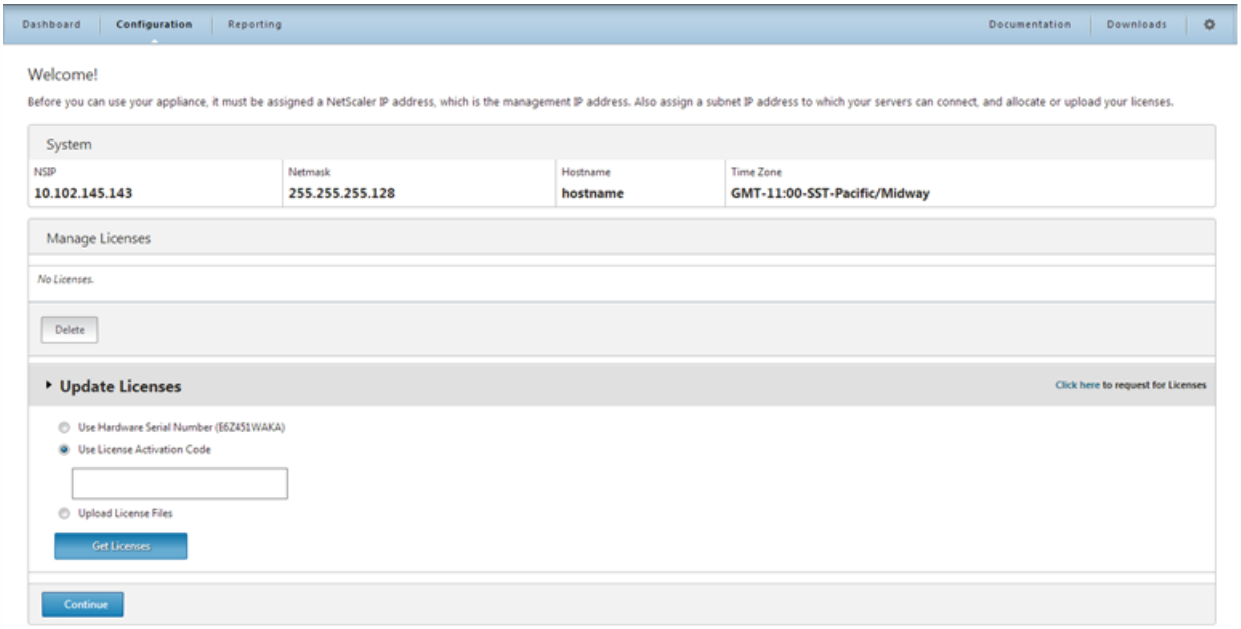


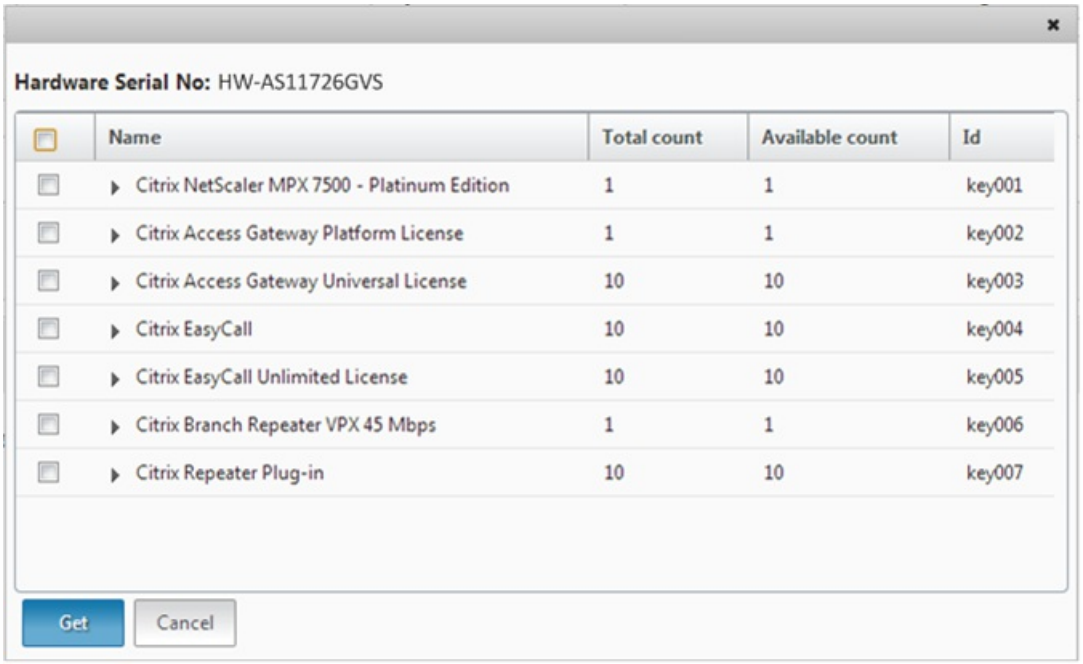


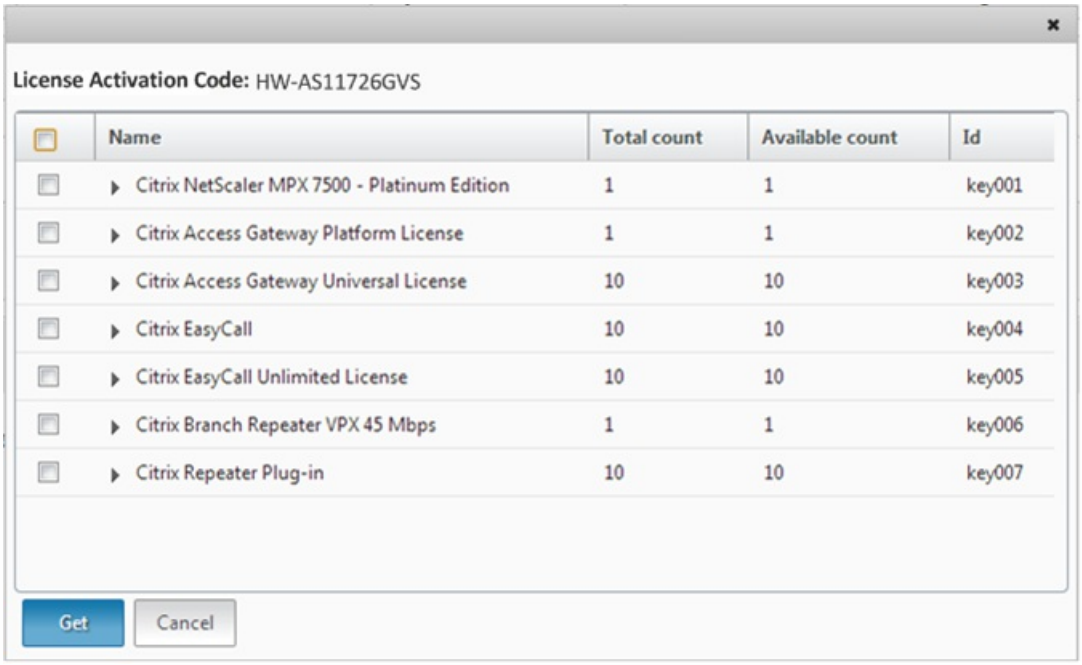


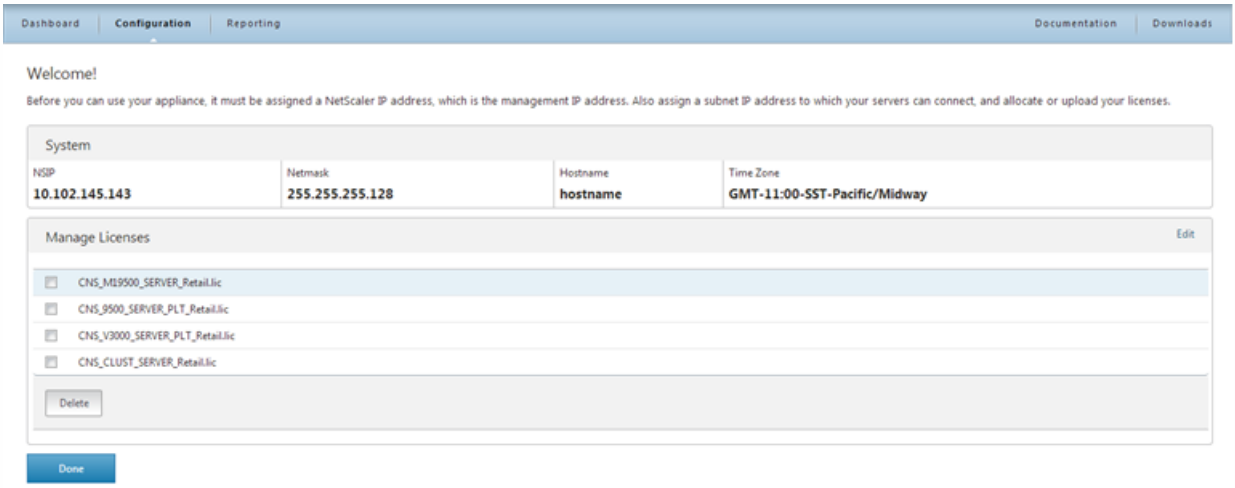


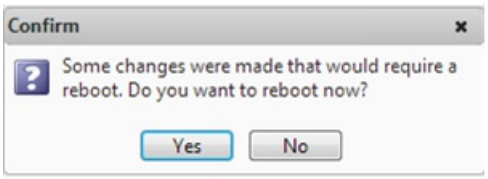
- 



- 





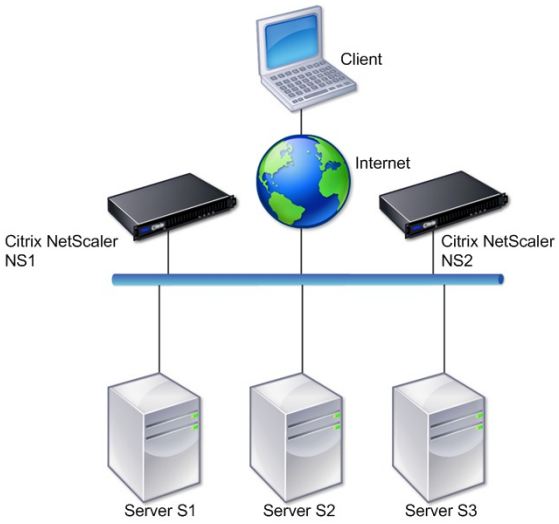


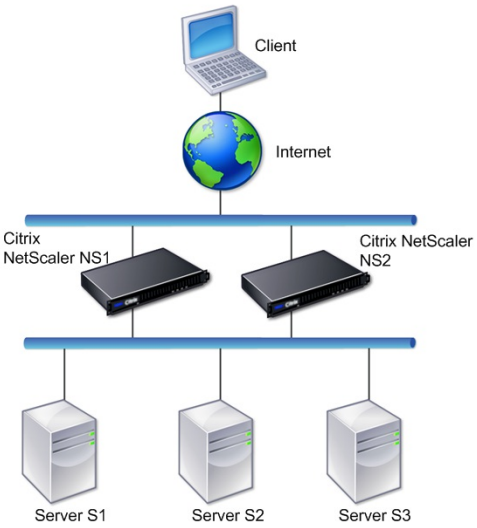
Invalid addr!
xxx.xxx.xxx.xxx

Exiting menu...
xxx.xxx.xxx.xxx

Values accepted,
Rebooting...

```
set ns config -ipaddress 10.102.29.60 -netmask 255.255.255.0
add ns ip 10.102.29.61 255.255.255.0 -type snip
add route 0.0.0.0 0.0.0.0 10.102.29.1
set system user nsroot -password
Enter password: *****
Confirm password: *****
save ns config
reboot
```





-
-

-
-

```
add HA node 0 10.102.29.170
 Done
```



```
> show HA node 0
1)    Node ID:    0
      IP:   10.102.29.200 (NS200)
      Node State: UP
      Master State: Primary
      SSL Card Status: UP
      Hello Interval: 200 msecs
      Dead Interval: 3 secs
      Node in this Master State for: 1:0:41:50 (days:hrs:min:sec)
```

-
-
-

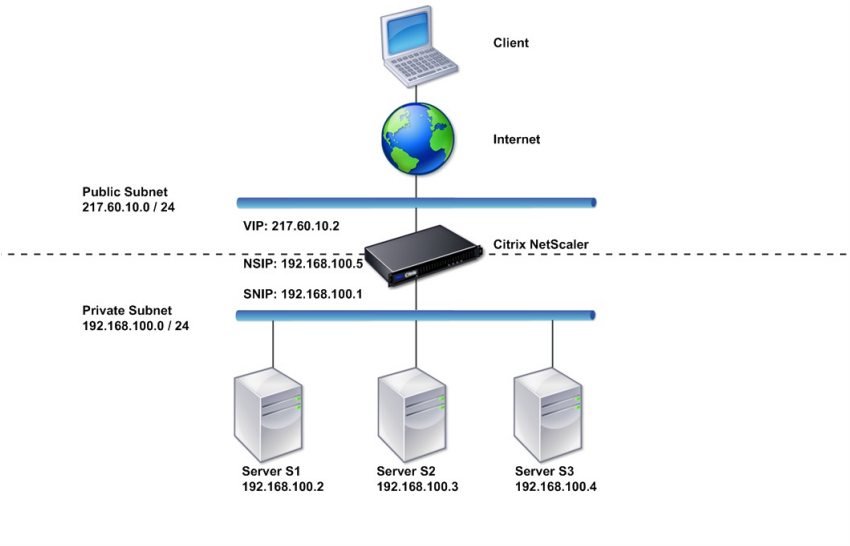
-
-

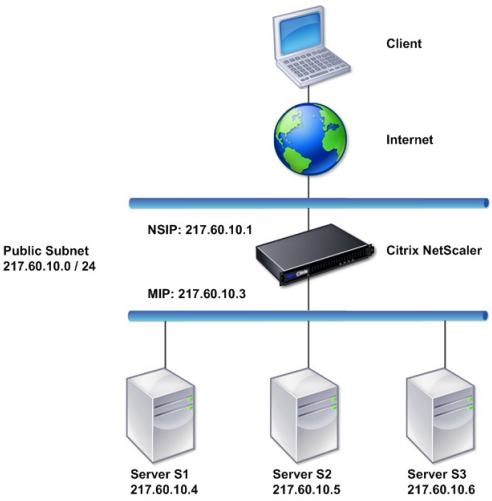
```
> set interface 1/8 -haMonitor OFF
 Done
```

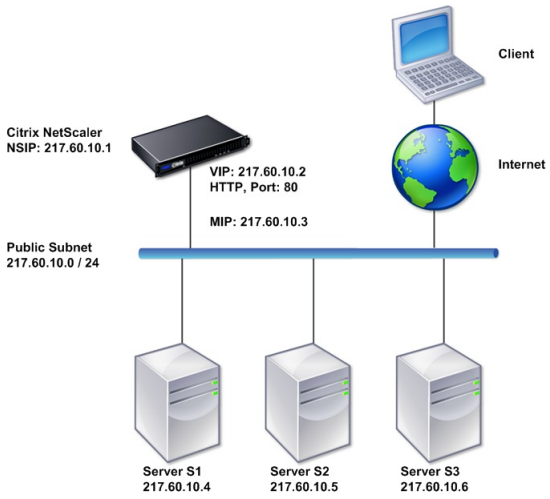


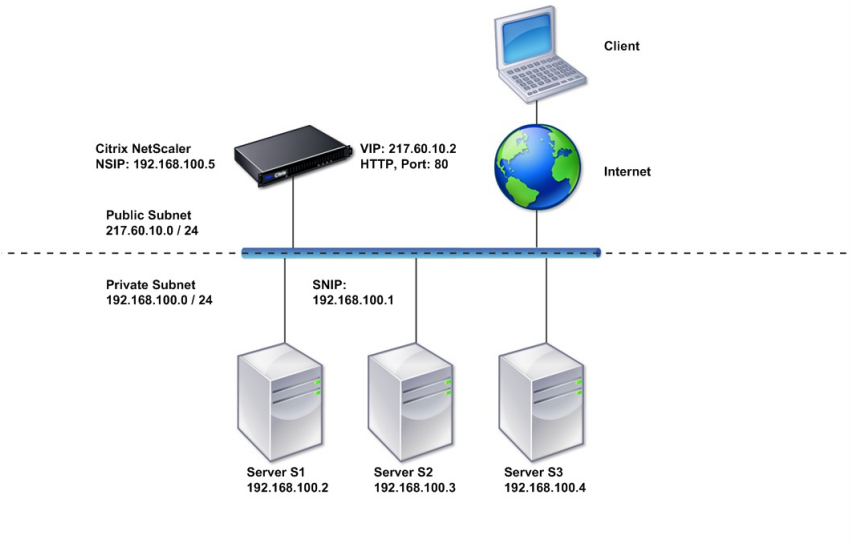
```
> show interface 1/8
    Interface 1/8 (Gig Ethernet 10/100/1000 MBits) #2
    flags=0x4000 <ENABLED, DOWN, down, autoneg, 802.1q>
    MTU=1514, native vlan=1, MAC=00:d0:68:15:fd:3d, downtime 238h55m44s
    Requested: media AUTO, speed AUTO, duplex AUTO, fctl OFF,
        throughput 0

    RX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
    TX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
    NIC: InDisc(0) OutDisc(0) Fctls(0) Stalls(0) Hangs(0) Muted(0)
    Bandwidth thresholds are not set.
```

-
-

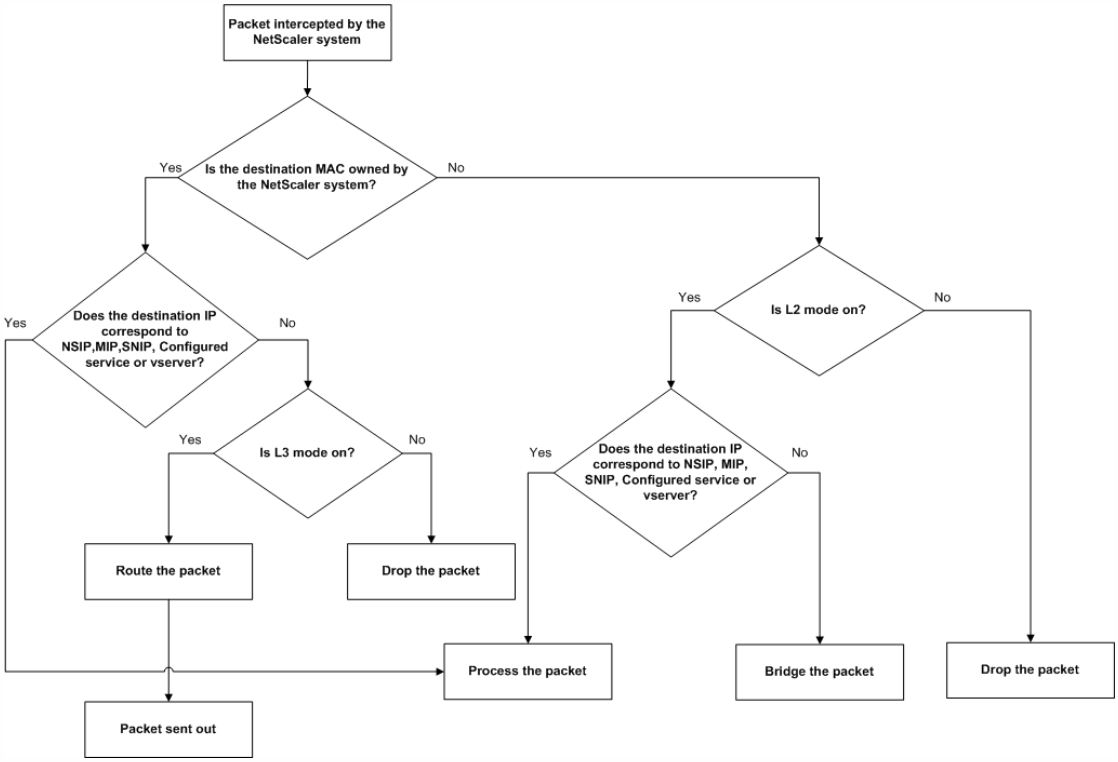








-
-
-
-
-
-
-



-
-
-

-
-
-

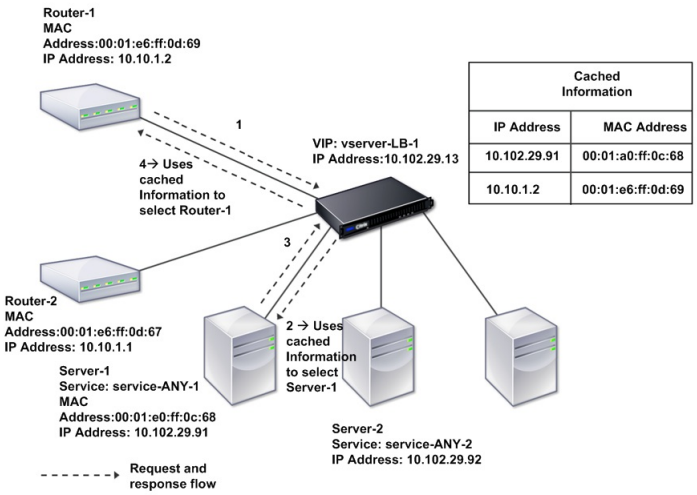
```
> enable ns mode l2
 Done
> show ns mode

    Mode              Acronym       Status
    -------           -------       ------
 1)  Fast Ramp          FR           ON
 2)  Layer 2 mode        L2           ON
 .
 .
 .
 Done
>

> disable ns mode l2
 Done
> show ns mode

    Mode              Acronym       Status
    -------           -------       ------
 1)  Fast Ramp          FR           ON
 2)  Layer 2 mode         L2           OFF
 .
 .
 .
 Done
>
```

-
-
-

```
> enable ns mode l3
 Done
> show ns mode

    Mode                  Acronym       Status
    -------               -------       ------
 1) Fast Ramp               FR            ON
 2) Layer 2 mode             L2            OFF
 .
 .
 .
 9) Layer 3 mode (ip forwarding)  L3           ON
 .
 .
 .
 Done
>

> disable ns mode l3
 Done
> show ns mode

    Mode                  Acronym       Status
    -------               -------       ------
```



```
1) Fast Ramp                FR          ON
2) Layer 2 mode              L2          OFF
.
.
.
9) Layer 3 mode (ip forwarding)  L3          OFF
.
.
.
Done
>
```

-
-

-
-
-

```
> enable ns mode mbf
 Done
```



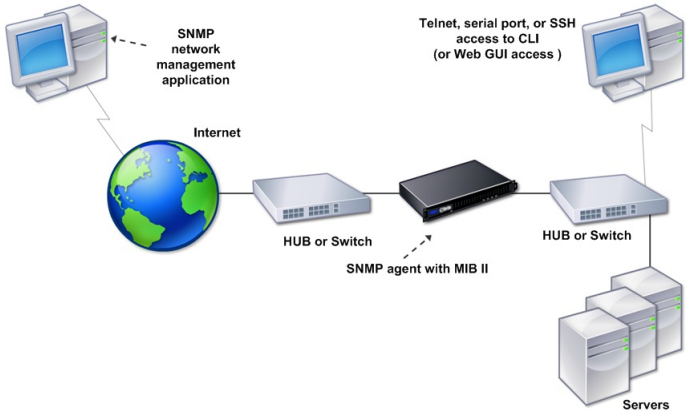
```
> show ns mode

   Mode                Acronym        Status
   -------             -------        ------
 1) Fast Ramp             FR           ON
 2) Layer 2 mode           L2           OFF
 .
 .
 .
 6) MAC-based forwarding      MBF          ON
 .
 .
 .
 Done
>

> disable ns mode mbf
 Done
> show ns mode

   Mode                Acronym        Status
   -------             -------        ------
 1) Fast Ramp             FR           ON
 2) Layer 2 mode           L2           OFF
 .
 .
 .
 6) MAC-based forwarding      MBF          OFF
 .
 .
 .
 Done
>
```

ntpdate -q <IP address or domain name of the NTP server>

/usr/sbin/ntpd -c /nsconfig/ntp.conf -l /var/log/ ntpd.log &

-
-

```
> add dns nameServer 10.102.29.10
 Done
> show dns nameServer 10.102.29.10
1)      10.102.29.10  -  State: DOWN
 Done
>
```



-
-

```
> add snmp manager 10.102.29.5 -netmask 255.255.255.255
 Done
> show snmp manager 10.102.29.5
1)     10.102.29.5        255.255.255.255
 Done
>
```

-
-

```
> add snmp trap specific 10.102.29.3
 Done
> show snmp trap
Type    DestinationIP   DestinationPort  Version  SourceIP     Min-Severity  Community
----    -------------   ---------------  -------  --------     ------------  ---------
generic  10.102.29.9    162          V2     NetScaler IP  N/A        public
generic  10.102.29.5    162          V2     NetScaler IP  N/A        public
generic  10.102.120.101  162          V2     NetScaler IP  N/A        public
.
.
.
specific 10.102.29.3    162          V2     NetScaler IP  -        public
 Done
>
```

-
-

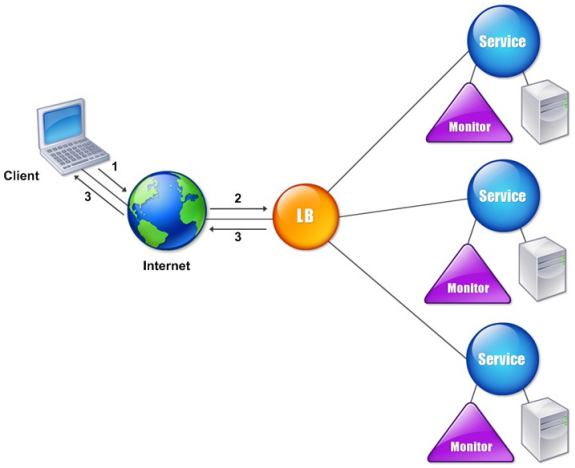
```
> set snmp alarm LOGIN-FAILURE -state ENABLED
 Done
> show snmp alarm LOGIN-FAILURE
  Alarm         Alarm Threshold  Normal Threshold  Time  State    Severity  Logging
  -----         ---------------  ----------------  ----  --------  ---------  --------
 1) LOGIN-FAILURE  N/A           N/A           N/A   ENABLED  -       ENABLED
 Done
>
```

-
-

```
> set snmp alarm LOGIN-FAILURE -severity Major
 Done
> show snmp alarm LOGIN-FAILURE
  Alarm         Alarm Threshold  Normal Threshold  Time  State    Severity  Logging
  -----         ---------------  ----------------  ----  --------  ---------  --------
 1) LOGIN-FAILURE  N/A           N/A           N/A   ENABLED  Major     ENABLED
 Done
>
```

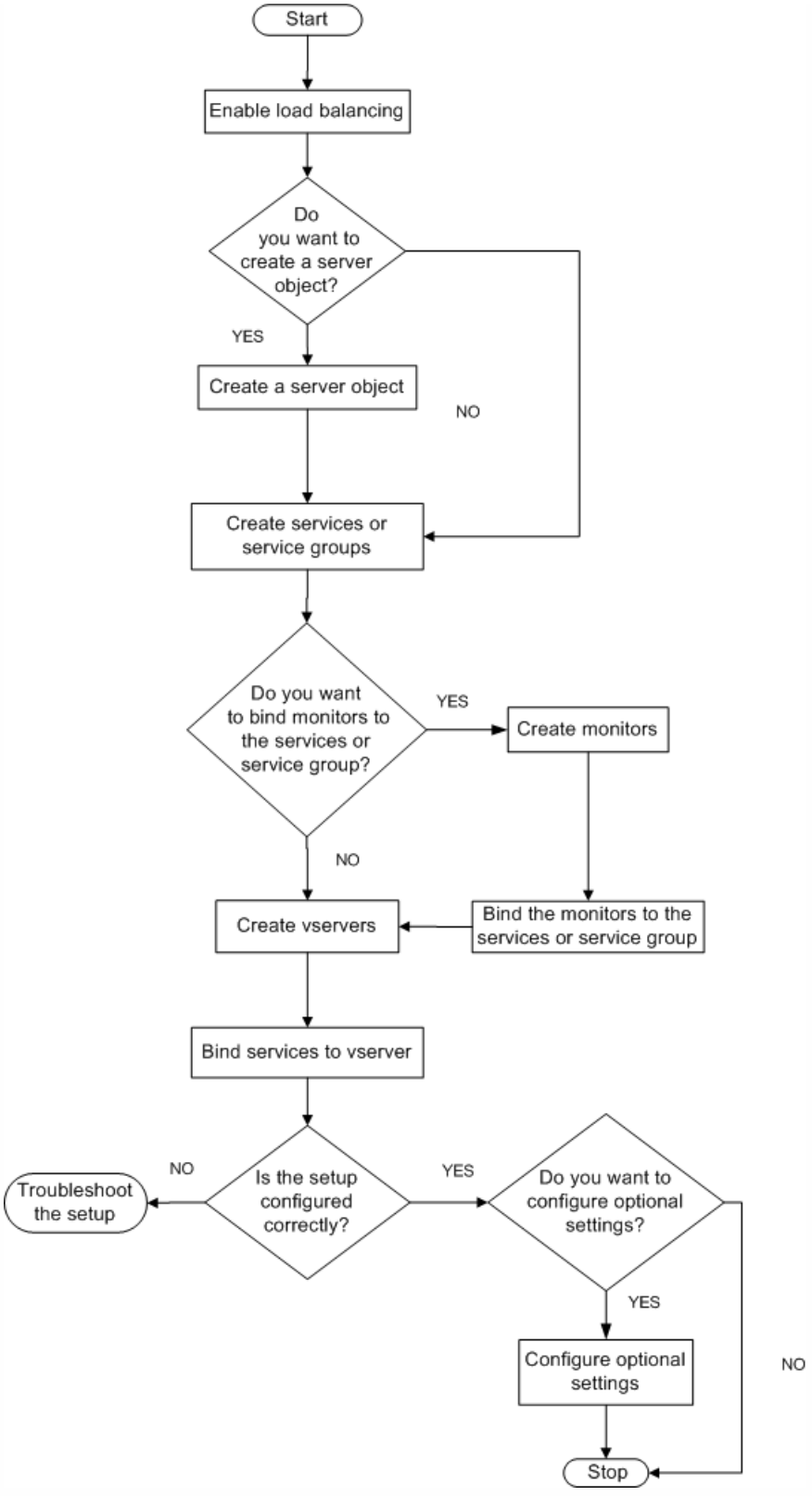

-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-

-
-
-
-
-



-
-

-
-



-
-

```
> enable feature lb
 Done
> show feature

    Feature              Acronym       Status
    -------              -------       ------
 1) Web Logging          WL            OFF
 2) Surge Protection     SP            OFF
 3) Load Balancing       LB            ON
 .
 .
 .
 9) SSL Offloading       SSL           ON
 .
 .
 .
Done
```

-

-
-
-

```
> add service service-HTTP-1 10.102.29.5 HTTP 80
 Done
> add lb vserver vserver-LB-1 HTTP 10.102.29.60 80
 Done
> bind lb vserver vserver-LB-1 service-HTTP-1
 Done
> show service bindings service-HTTP-1
   service-HTTP-1 (10.102.29.5:80) - State : DOWN

      1)     vserver-LB-1 (10.102.29.60:80) - State : DOWN
 Done
```


-
-
-

-

-

-
-

```
> set lb vserver vserver-LB-1 -persistenceType COOKIEINSERT
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   .
   .
   .
   Persistence: COOKIEINSERT (version 0)   Persistence Timeout: 2 min
   .
   .
   .
 Done
>
```

-
-

```
> set lb vserver vserver-LB-1 -persistenceType URLPASSIVE
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   .
   .
   .
   Persistence: URLPASSIVE Persistence Timeout: 2 min
   .
   .
   .
```

```
 Done
>
```

-
-

```
> set lb vserver vserver-LB-1 -redirectURL http://www.newdomain.com/mysite/maintenance
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   State: DOWN
   Last state change was at Wed Jun 17 08:56:34 2009 (+666 ms)
   .
   .
   .
   Redirect URL: http://www.newdomain.com/mysite/maintenance
   .
   .
   .
 Done
```

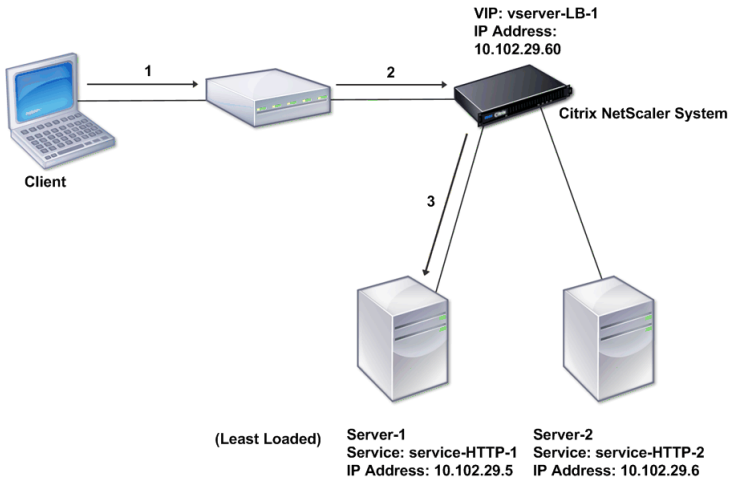


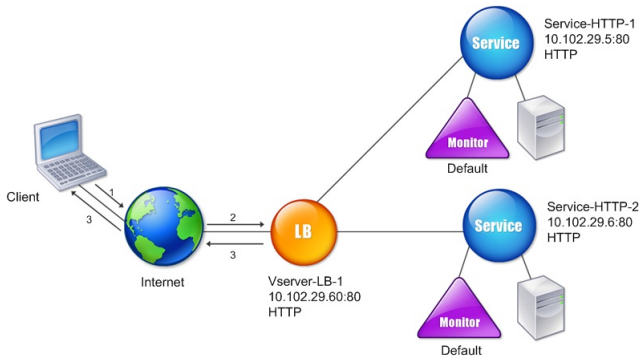
```
>
```

-
-

```
> set lb vserver vserver-LB-1 -backupVserver vserver-LB-2
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   State: DOWN
```

```
    Last state change was at Wed Jun 17 08:56:34 2009 (+661 ms)
    .
    .
    .
    Backup: vserver-LB-2
    .
    .
    .
 Done
>
```





Accelerating Load Balanced Traffic by Using Compression

Aug 22, 2013

Compression is a popular means of optimizing bandwidth usage, and most web browsers support compressed data. If you enable the compression feature, the NetScaler appliance intercepts requests from clients and determines whether the client can accept compressed content. After receiving the HTTP response from the server, the appliance examines the content to determine whether it is compressible. If the content is compressible, the appliance compresses it, modifies the response header to indicate the type of compression performed, and forwards the compressed content to the client.

NetScaler compression is a policy-based feature. A policy filters requests and responses to identify responses to be compressed, and specifies the type of compression to apply to each response. The appliance provides several built-in policies to compress common MIME types such as text/html, text/plain, text/xml, text/css, text/rtf, application/msword, application/vnd.ms-excel, and application/vnd.ms-powerpoint. You can also create custom policies. The appliance does not compress compressed MIME types such as application/octet-stream, binary, bytes, and compressed image formats such as GIF and JPEG.

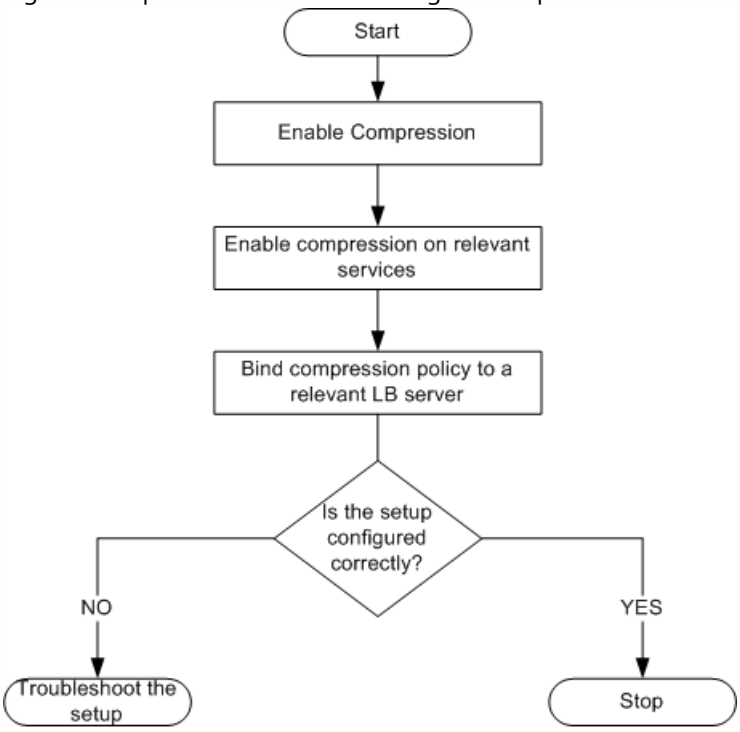
To configure compression, you must enable it globally and on each service that will provide responses that you want compressed. If you have configured virtual servers for load balancing or content switching, you should bind the polices to the virtual servers. Otherwise, the policies apply to all traffic that passes through the appliance.

Compression Configuration Task Sequence

Updated: 2013-08-22

The following flow chart shows the sequence of tasks for configuring basic compression in a load balancing setup.

Figure 1. Sequence of Tasks to Configure Compression



Note: The steps in the above figure assume that load balancing has already been configured.

Enabling Compression

Updated: 2013-06-07

By default, compression is not enabled. You must enable the compression feature to allow compression of HTTP responses that are sent to the client.

To enable compression by using the command line interface

At the command prompt, type the following commands to enable compression and verify the configuration:

- enable ns feature CMP
- show ns feature

Example

```
> enable ns feature CMP
Done
> show ns feature

     Feature                Acronym          Status
     -------                -------          ------
1)   Web Logging             WL               ON
2)   Surge Protection        SP               OFF
.
7)   Compression Control     CMP              ON
8)   Priority Queuing        PQ               OFF
.
Done
```

To enable compression by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Compression check box, and then click OK.
4. In the Enable/Disable Feature(s)? dialog box, click Yes.

Configuring Services to Compress Data

Updated: 2013-08-22

In addition to enabling compression globally, you must enable it on each service that will deliver files to be compressed.

To enable compression on a service by using the command line

At the command prompt, type the following commands to enable compression on a service and verify the configuration:

- set service <name> -CMP YES
- show service <name>

Example

```
> show service SVC_HTTP1
SVC_HTTP1 (10.102.29.18:80) - HTTP
State: UP
Last state change was at Tue Jun 16 06:19:14 2009 (+737 ms)
Time since last state change: 0 days, 03:03:37.200
Server Name: 10.102.29.18
Server ID : 0   Monitor Threshold : 0
Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
Use Source IP: NO
Client Keepalive(CKA): NO
Access Down Service: NO
TCP Buffering(TCPB): NO
HTTP Compression(CMP): YES
Idle timeout: Client: 180 sec   Server: 360 sec
Client IP: DISABLED
Cacheable: NO
SC: OFF
SP: OFF
Down state flush: ENABLED

1)      Monitor Name: tcp-default
State: DOWN     Weight: 1
Probes: 1095    Failed [Total: 1095 Current: 1095]
Last response: Failure - TCP syn sent, reset received.
Response Time: N/A
Done
```

To enable compression on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure compression (for example, service-HTTP-1), and then click Open.
3. On the Advanced tab, under Settings, select the Compression check box, and then click OK.
4. Verify that, when the service is selected, HTTP Compression(CMP): ON appears in the **Details** section at the bottom of the pane.

Binding a Compression Policy to a Virtual Server

Updated: 2013-09-04

If you bind a policy to a virtual server, the policy is evaluated only by the services associated with that virtual server. You can bind compression policies to a virtual server either from the Configure Virtual Server (Load Balancing) dialog box or from the Compression Policy Manager dialog box. This topic includes instructions to bind compression policies to a load balancing virtual server by using the Configure Virtual Server (Load Balancing) dialog box. For information about how you can bind a compression policy to a load balancing virtual server by using the Compression Policy Manager dialog box, see "Configuring and Binding Policies with the Policy Manager."

To bind or unbind a compression policy to a virtual server by using the command line

At the command prompt, type the following commands to bind or unbind a compression policy to a load balancing virtual server and verify the configuration:

- (bind | unbind) lb vserver <name> -policyName <string>
- show lb vserver <name>

Example

```
> bind lb vserver lbvip -policyName ns_cmp_msapp
Done
> show lb vserver lbvip
lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
State: UP
Last state change was at Thu May 28 05:37:21 2009 (+685 ms)
Time since last state change: 19 days, 04:26:50.470
Effective State: UP
Client Idle Timeout: 180 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
Port Rewrite : DISABLED
No. of Bound Services :  1 (Total)      1 (Active)
Configured Method: LEASTCONNECTION
Current Method: Round Robin, Reason: Bound service's state changed to UP
Mode: IP
Persistence: NONE
Vserver IP and Port insertion: OFF
Push: DISABLED  Push VServer:
Push Multi Clients: NO
Push Label Rule:

Bound Service Groups:
1)      Group Name: Service-Group-1

1) Service-Group-1 (10.102.29.252: 80) - HTTP State: UP Weight:                 1

1)      Policy : ns_cmp_msapp Priority:0
Done
```

To bind or unbind a compression policy to a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind or unbind a compression policy (for example, Vserver-LB-1), and then click Open.

3. In the Configure Virtual Server (Load Balancing) dialog box, on the Policies tab, click Compression.
4. Do one of the following:
   - To bind a compression policy, click Insert Policy, and then select the policy you want to bind to the virtual server.
   - To unbind a compression policy, click the name of the policy you want to unbind from the virtual server, and then click Unbind Policy.
5. Click OK.

Securing Load Balanced Traffic by Using SSL

Jan 31, 2011

The Citrix NetScaler SSL offload feature transparently improves the performance of web sites that conduct SSL transactions. By offloading CPU-intensive SSL encryption and decryption tasks from the local web server to the appliance, SSL offloading ensures secure delivery of web applications without the performance penalty incurred when the server processes the SSL data. Once the SSL traffic is decrypted, it can be processed by all standard services. The SSL protocol works seamlessly with various types of HTTP and TCP data and provides a secure channel for transactions using such data.

To configure SSL, you must first enable it. Then, you configure HTTP or TCP services and an SSL virtual server on the appliance, and bind the services to the virtual server. You must also add a certificate-key pair and bind it to the SSL virtual server. If you use Outlook Web Access servers, you must create an action to enable SSL support and a policy to apply the action. An SSL virtual server intercepts incoming encrypted traffic and decrypts it by using a negotiated algorithm. The SSL virtual server then forwards the decrypted data to the other entities on the appliance for appropriate processing.

This document includes the following:

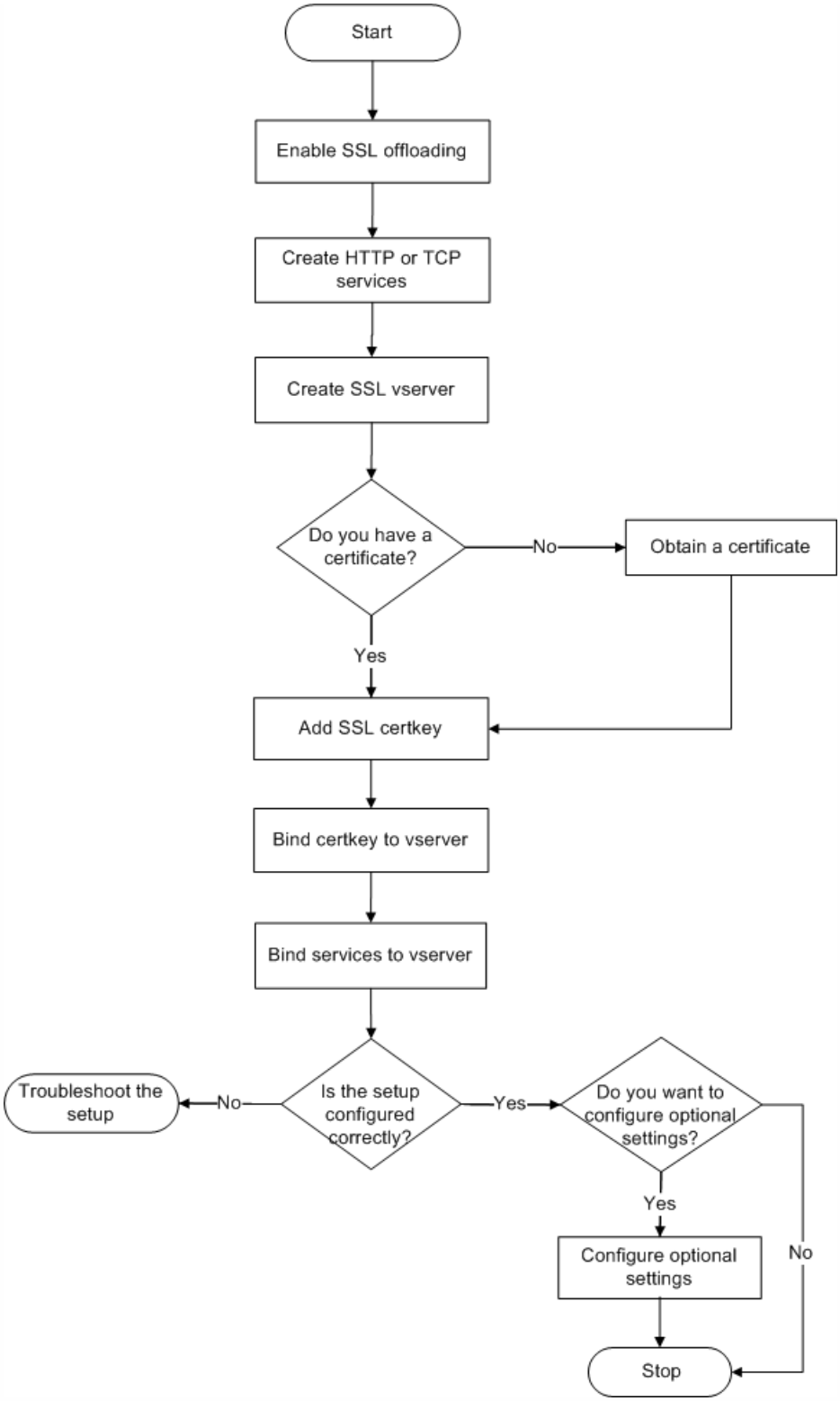
SSL Configuration Task Sequence

To configure SSL, you must first enable it. Then, you must create an SSL virtual server and HTTP or TCP services on the NetScaler. Finally, you must bind a valid SSL certificate and the configured services to the SSL virtual server.

An SSL virtual server intercepts incoming encrypted traffic and decrypts it using a negotiated algorithm. The SSL virtual server then forwards the decrypted data to the other entities on the NetScaler for appropriate processing.

The following flow chart shows the sequence of tasks for configuring a basic SSL offload setup.

Figure 1. Sequence of Tasks to Configure SSL Offloading



Enabling SSL Offload

Updated: 2013-06-05

You should enable the SSL feature before configuring SSL offload. You can configure SSL-based entities on the appliance without enabling the SSL feature, but they will not work until you enable SSL.

To enable SSL by using the command line interface

At the command prompt, type the following commands to enable SSL Offload and verify the configuration:

- enable ns feature SSL
- show ns feature

Example

```
> enable ns feature ssl
Done
> show ns feature
Feature Acronym Status
------- ------- ------
1) Web Logging WL ON
2) SurgeProtection SP OFF
3) Load Balancing LB ON . . .
 9) SSL Offloading SSL ON
10) Global Server Load Balancing GSLB ON . .
Done >
```

To enable SSL by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. Select the SSL Offloading check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes.

Creating HTTP Services

Updated: 2013-08-23

A service on the appliance represents an application on a server. Once configured, services are in the disabled state until the appliance can reach the server on the network and monitor its status. This topic covers the steps to create an HTTP service.

Note: For TCP traffic, perform the procedures in this and the following topics, but create TCP services instead of HTTP services.

To add an HTTP service by using the command line interface

At the command prompt, type the following commands to add a HTTP service and verify the configuration:

- add service <name> (<IP> | <serverName>) <serviceType> <port>
- show service <name>

```
> add service SVC_HTTP1 10.102.29.18 HTTP 80
 Done
> show service SVC_HTTP1
      SVC_HTTP1 (10.102.29.18:80) - HTTP
      State: UP
      Last state change was at Wed Jul 15 06:13:05 2009
```



```
        Time since last state change: 0 days, 00:00:15.350
        Server Name: 10.102.29.18
        Server ID : 0   Monitor Threshold : 0
        Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
        Use Source IP: NO
        Client Keepalive(CKA): NO
        Access Down Service: NO
        TCP Buffering(TCPB): NO
        HTTP Compression(CMP): YES
        Idle timeout: Client: 180 sec   Server: 360 sec
        Client IP: DISABLED
        Cacheable: NO
        SC: OFF
        SP: OFF
        Down state flush: ENABLED

    1)    Monitor Name: tcp-default
              State: UP       Weight: 1
              Probes: 4       Failed [Total: 0 Current: 0]
              Last response: Success - TCP syn+ack received.
              Response Time: N/A
    Done
```

To add an HTTP service by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Services.
2. In details pane, click Add.
3. In the Create Service dialog box, in the Service Name, Server, and Port text boxes, type the name of the service, IP address, and port (for example, SVC_HTTP1, 10.102.29.18, and 80).
4. In the Protocol list, select the type of the service (for example, HTTP).
5. Click Create, and then click Close. The HTTP service you configured appears in the Services page.
6. Verify that the parameters you configured are correctly configured by selecting the service and viewing the Details section at the bottom of the pane.

Adding an SSL-Based Virtual Server

Updated: 2013-06-05

In a basic SSL offloading setup, the SSL virtual server intercepts encrypted traffic, decrypts it, and sends the clear text messages to the services that are bound to the virtual server. Offloading CPU-intensive SSL processing to the appliance allows the back-end servers to process a greater number of requests.

To add an SSL-based virtual server by using the command line interface

At the command prompt, type the following commands to create an SSL-based virtual server and verify the configuration:

- add lb vserver <name> <serviceType> [<IPAddress> <port>]
- show lb vserver <name>

Example

```
> add lb vserver vserver-SSL-1 SSL 10.102.29.50 443
Done
> show lb vserver vserver-SSL-1
vserver-SSL-1 (10.102.29.50:443) - SSL Type: ADDRESS
State: DOWN[Certkey not bound] Last state change was at Tue Jun 16 06:33:08 2009 (+176 ms)
Time since last state change: 0 days, 00:03:44.120
Effective State: DOWN Client Idle Timeout: 180 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
No. of Bound Services : 0 (Total) 0 (Active)
Configured Method: LEASTCONNECTION Mode: IP
Persistence: NONE
Vserver IP and Port insertion: OFF
Push: DISABLED Push VServer: Push Multi Clients: NO Push Label Rule: Done
```

Caution: To ensure secure connections, you must bind a valid SSL certificate to the SSL-based virtual server before you enable it.

To add an SSL-based virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (SSL Offload) dialog box, in the Name, IP Address, and Port text boxes, type the name of the virtual server, IP address, and port (for example, Vserver-SSL-1, 10.102.29.50, and 443).
4. In the Protocol list, select the type of the virtual server, for example, SSL.
5. Click Create, and then click Close.
6. Verify that the parameters you configured are correctly configured by selecting the virtual server and viewing the Details section at the bottom of the pane. The virtual server is marked as DOWN because a certificate-key pair and services have not been bound to it.

Caution: To ensure secure connections, you must bind a valid SSL certificate to the SSL-based virtual server before you enable it.

Binding Services to the SSL Virtual Server

Updated: 2013-08-23

After decrypting the incoming data, the SSL virtual server forwards the data to the services that you have bound to the virtual server.

Data transfer between the appliance and the servers can be encrypted or in clear text. If the data transfer between the appliance and the servers is encrypted, the entire transaction is secure from end to end. For more information about configuring the system for end-to-end security, see "SSL Offload and Acceleration."

To bind a service to a virtual server by using the command line interface

At the command prompt, type the following commands to bind service to the SSL virtual server and verify the configuration:

- bind lb vserver <name> <serviceName>
- show lb vserver <name>

Example

```
> bind lb vserver vserver-SSL-1 SVC_HTTP1
Done
> show lb vserver vserver-SSL-1 vserver-SSL-1 (10.102.29.50:443) - SSL Type:
ADDRESS State: DOWN[Certkey not bound]
Last state change was at Tue Jun 16 06:33:08 2009 (+174 ms)
Time since last state change: 0 days, 00:31:53.70
Effective State: DOWN Client Idle
Timeout: 180 sec
Down state flush: ENABLED Disable Primary Vserver On Down :
DISABLED No. of Bound Services : 1 (Total) 0 (Active)
Configured Method: LEASTCONNECTION Mode: IP Persistence: NONE Vserver IP and
Port insertion: OFF Push: DISABLED Push VServer: Push Multi Clients: NO Push Label Rule:

1) SVC_HTTP1 (10.102.29.18: 80) - HTTP
State: DOWN Weight: 1
Done
```

To bind a service to a virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, select a virtual server, and then click Open.
3. On the Services tab, in the Active column, select the check boxes next to the services that you want to bind to the selected virtual server.
4. Click OK.
5. Verify that the Number of Bound Services counter in the Details section at the bottom of the pane is incremented by the number of services that you bound to the virtual server.

Adding a Certificate Key Pair

Updated: 2013-06-24

An SSL certificate is an integral element of the SSL Key-Exchange and encryption/decryption process. The certificate is used during SSL handshake to establish the identity of the SSL server. You can use a valid, existing SSL certificate that you have on the NetScaler appliance, or you can create your own SSL certificate. The appliance supports RSA/DSA certificates of up to 4096 bits.

Note: Citrix recommends that you use a valid SSL certificate that has been issued by a trusted certificate authority. Invalid certificates and self-created certificates are not compatible with all SSL clients.
Before a certificate can be used for SSL processing, you must pair it with its corresponding key. The certificate key pair is then bound to the virtual server and used for SSL processing.

To add a certificate key pair by using the command line interface

At the command prompt, type the following commands to create a certificate key pair and verify the configuration:

- add ssl certKey <certkeyName> -cert <string> [-key <string>]
- show sslcertkey <name>

Example

```
> add ssl certKey CertKey-SSL-1 -cert ns-root.cert -key ns-root.key
 Done
> show sslcertkey CertKey-SSL-1
   Name: CertKey-SSL-1 Status: Valid,
   Days to expiration:4811 Version: 3
   Serial Number: 00 Signature Algorithm: md5WithRSAEncryption Issuer: C=US,ST=California,L=San
   Jose,O=Citrix ANG,OU=NS Internal,CN=de fault
   Validity Not Before: Oct 6 06:52:07 2006 GMT Not After : Aug 17 21:26:47 2022 GMT
   Subject: C=US,ST=California,L=San Jose,O=Citrix ANG,OU=NS Internal,CN=d efault Public Key
   Algorithm: rsaEncryption Public Key
   size: 1024
 Done
```

To add a certificate key pair by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. In the details pane, click Add.
3. In the Install Certificate dialog box, in the Certificate-Key Pair Name text box, type a name for the certificate key pair you want to add, for example, Certkey-SSL-1.
4. Under Details, in Certificate File Name, click Browse (Appliance) to locate the certificate. Both the certificate and the key are stored in the /nsconfig/ssl/ folder on the appliance. To use a certificate present on the local system, select Local.
5. Select the certificate you want to use, and then click Select.
6. In Private Key File Name, click Browse (Appliance) to locate the private key file. To use a private key present on the local system, select Local.
7. Select the key you want to use and click Select. To encrypt the key used in the certificate key pair, type the password to be used for encryption in the Password text box.
8. Click Install.
9. Double-click the certificate key pair and, in the Certificate Details window, verify that the parameters have been configured correctly and saved.

Binding an SSL Certificate Key Pair to the Virtual Server

Updated: 2013-06-24

After you have paired an SSL certificate with its corresponding key, you must bind the certificate key pair to the SSL virtual server so that it can be used for SSL processing. Secure sessions require establishing a connection between the client computer and an SSL-based virtual server on the appliance. SSL processing is then carried out on the incoming traffic at the virtual server. Therefore, before enabling the SSL virtual server on the appliance, you need to bind a valid SSL certificate to the SSL virtual server.

To bind an SSL certificate key pair to a virtual server by using the command line

interface

At the command prompt, type the following commands to bind an SSL certificate key pair to a virtual server and verify the configuration:

- bind ssl vserver <vServerName> -certkeyName <string>
- show ssl vserver <name>

Example

```
 > bind ssl vserver Vserver-SSL-1 -certkeyName CertKey-SSL-1
Done
> show ssl vserver Vserver-SSL-1

    Advanced SSL configuration for VServer Vserver-SSL-1:
    DH: DISABLED
    Ephemeral RSA: ENABLED Refresh Count: 0
    Session Reuse: ENABLED Timeout: 120 seconds
    Cipher Redirect: ENABLED
    SSLv2 Redirect: ENABLED
    ClearText Port: 0
    Client Auth: DISABLED
    SSL Redirect: DISABLED
    Non FIPS Ciphers: DISABLED
    SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED

1) CertKey Name: CertKey-SSL-1 Server Certificate
1) Cipher Name: DEFAULT
  Description: Predefined Cipher Alias
Done
```

To bind an SSL certificate key pair to a virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server to which you want to bind the certificate key pair, for example, Vserver-SSL-1, and click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, on the SSL Settings tab, under Available, select the certificate key pair that you want to bind to the virtual server (for example, Certkey-SSL-1), and then click Add.
4. Click OK.
5. Verify that the certificate key pair that you selected appears in the Configured area.

Configuring Support for Outlook Web Access

If you use Outlook Web Access (OWA) servers on your NetScaler appliance, you must configure the appliance to insert a special header field, FRONT-END-HTTPS: ON, in HTTP requests directed to the OWA servers, so that the servers generate URL links as https:// instead of http://.

Note: You can enable OWA support for HTTP-based SSL virtual servers and services only. You cannot apply it for TCP-based SSL virtual servers and services.
To configure OWA support, do the following:

- Create an SSL action to enable OWA support.
- Create an SSL policy.
- Bind the policy to the SSL virtual server.

Creating an SSL Action to Enable OWA Support

Updated: 2013-06-24

Before you can enable Outlook Web Access (OWA) support, you must create an SSL action. SSL actions are bound to SSL policies and triggered when incoming data matches the rule specified by the policy.

To create an SSL action to enable OWA support by using the command line interface

At the command prompt, type the following commands to create an SSL action to enable OWA support and verify the configuration:

- add ssl action <name> -OWASupport ENABLED
- show SSL action <name>

```
 > add ssl action Action-SSL-OWA -OWASupport enabled
Done
> show SSL action Action-SSL-OWA
  Name: Action-SSL-OWA
  Data Insertion Action: OWA
  Support: ENABLED
Done
```

To create an SSL action to enable OWA support by using the configuration utility

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, on the Actions tab, click Add.
3. In the Create SSL Action dialog box, in the Name text box, type Action-SSL-OWA.
4. Under Outlook Web Access, select Enabled.
5. Click Create, and then click Close.
6. Verify that Action-SSL-OWA appears in the **SSL Actions** page.

Creating SSL Policies

Updated: 2013-09-04

SSL policies are created by using the policy infrastructure. Each SSL policy has an SSL action bound to it, and the action is carried out when incoming traffic matches the rule that has been configured in the policy.

To create an SSL policy by using the command line interface

At the command prompt, type the following commands to configure an SSL policy and verify the configuration:

- add ssl policy <name> -rule <expression> -reqAction <string>
- show ssl policy <name>

Example

```
> add ssl policy Policy-SSL-1 -rule ns_true -reqaction Action-SSL-OWA
```



```
Done
> show ssl policy Policy-SSL-1
Name: Policy-SSL-1      Rule: ns_true
Action: Action-SSL-OWA  Hits: 0
Policy is bound to following entities
1)      PRIORITY : 0
Done
```

To create an SSL policy by using the configuration utility

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, click Add.
3. In the Create SSL Policy dialog box, in the Name text box, type the name of the SSL Policy (for example, Policy-SSL-1).
4. In Request Action, select the configured SSL action that you want to associate with this policy (for example, Action-SSL-OWA). The ns_true general expression applies the policy to all successful SSL handshake traffic. However, if you need to filter specific responses, you can create policies with a higher level of detail. For more information about configuring granular policy expressions, see "Understanding Policies and Expressions."
5. In Named Expressions, choose the built-in general expression ns_true and click Add Expression. The expression ns_true now appears in the Expression text box.
6. Click Create, and then click Close.
7. Verify that the policy is correctly configured by selecting the policy and viewing the Details section at the bottom of the pane.

Binding the SSL Policy to an SSL Virtual Server

Updated: 2013-06-24

After you configure an SSL policy for Outlook Web Access, bind the policy to a virtual server that will intercept incoming Outlook traffic. If the incoming data matches any of the rules configured in the SSL policy, the policy is triggered and the action associated with it is carried out.

To bind an SSL policy to an SSL virtual server by using the command line interface

At the command prompt, type the following commands to bind an SSL policy to an SSL virtual server and verify the configuration:

- bind ssl vserver <vServerName> -policyName <string>
- show ssl vserver <name>

Example

```
> bind ssl vserver Vserver-SSL-1 -policyName Policy-SSL-1
 Done
> show ssl vserver Vserver-SSL-1
      Advanced SSL configuration for VServer Vserver-SSL-1:
      DH: DISABLED
      Ephemeral RSA: ENABLED        Refresh Count: 0
      Session Reuse: ENABLED        Timeout: 120 seconds
      Cipher Redirect: ENABLED
```

```
        SSLv2 Redirect: ENABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED

    1)    CertKey Name: CertKey-SSL-1 Server Certificate

    1)    Policy Name: Policy-SSL-1
                    Priority: 0
    1)    Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
     Done
    >
```

To bind an SSL policy to an SSL virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, select the virtual server (for example, Vserver-SSL-1), and then click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, click Insert Policy, and then select the policy that you want to bind to the SSL virtual server. Optionally, you can double-click the Priority field and type a new priority level.
4. Click OK.

Features at a Glance

Sep 04, 2013

Citrix NetScaler features can be configured independently or in combinations to address specific needs. Although some features fit more than one category, the numerous NetScaler features can generally be categorized as application switching and traffic management features, application acceleration features, and application security and firewall features, and an application visibility feature.

To understand the order in which the features perform their processing, see "Processing Order of Features."

This document includes the following:

Application Switching and Traffic Management Features

Aug 30, 2016

SSL Offloading

Transparently offloads SSL encryption and decryption from web servers, freeing server resources to service content requests. SSL places a heavy burden on an application's performance and can render many optimization measures ineffective. SSL offload and acceleration allow all the benefits of Citrix Request Switching technology to be applied to SSL traffic, ensuring secure delivery of web applications without degrading end-user performance.
For more information, see "SSL Offload and Acceleration."

Access Control Lists

Compares incoming packets to Access Control Lists (ACLs). If a packet matches an ACL rule, the action specified in the rule is applied to the packet. Otherwise, the default action (ALLOW) is applied and the packet is processed normally. For the appliance to compare incoming packets to the ACLs, you have to apply the ACLs. All ACLs are enabled by default, but you have to apply them in order for the NetScaler to compare incoming packets against them. If an ACL is not required to be a part of the lookup table, but still needs to be retained in the configuration, it should be disabled before the ACLs are applied. A NetScaler does not compare incoming packets to disabled ACLs.
For more information, see "Access Control List."

Load Balancing

Load balancing decisions are based on a variety of algorithms, including round robin, least connections, weighted least bandwidth, weighted least packets, minimum response time, and hashing based on URL, domain source IP, or destination IP. Both the TCP and UDP protocols are supported, so the NetScaler can load balance all traffic that uses those protocols as the underlying carrier (for example, HTTP, HTTPS, UDP, DNS, NNTP, and general firewall traffic). In addition, the NetScaler can maintain session persistence based on source IP, cookie, server, group, or SSL session. It allows users to apply custom Extended Content Verification (ECV) to servers, caches, firewalls and other infrastructure devices to ensure that these systems are functioning properly and are providing the right content to users. It can also perform health checks using ping, TCP, or HTTP URL, and the user can create monitors based on Perl scripts. To provide high-scale WAN optimization, the CloudBridge appliances deployed at data centers can be load balanced through NetScaler appliances. The bandwidth and number of concurrent sessions can be improved significantly.
For more information, see "Load Balancing."

Traffic Domains

Traffic domains provide a way to create logical ADC partitions within a single NetScaler appliance. They enable you to segment network traffic for different applications. You can use traffic domains to create multiple isolated environments whose resources do not interact with each other. An application belonging to a specific traffic domain communicates only with entities, and processes traffic, within that domain. Traffic belonging to one traffic domain cannot cross the boundary of another traffic domain. Therefore, you can use duplicate IP addresses on the appliance as long as an addresses is not duplicated within the same domain.
For more information, see "Traffic Domains."

Network Address Translation

Network address translation (NAT) involves modification of the source and/or destination IP addresses, and/or the TCP/UDP port numbers, of IP packets that pass through the NetScaler appliance. Enabling NAT on the appliance enhances

the security of your private network, and protects it from a public network such as the Internet, by modifying your network's source IP addresses when data passes through the NetScaler.

The NetScaler appliance supports the following types of network address translation:

INAT—In Inbound NAT (INAT), an IP address (usually public) configured on the NetScaler appliance listens to connection requests on behalf of a server. For a request packet received by the appliance on a public IP address, the NetScaler replaces the destination IP address with the private IP address of the server. In other words, the appliance acts as a proxy between clients and the server. INAT configuration involves INAT rules, which define a 1:1 relationship between the IP address on the NetScaler appliance and the IP address of the server.

RNAT—In Reverse Network Address Translation (RNAT), for a session initiated by a server, the NetScaler appliance replaces the source IP address in the packets generated by the server with an IP address (type SNIP) configured on the appliance. The appliance thereby prevents exposure of the server's IP address in any of the packets generated by the server. An RNAT configuration involves an RNAT rule, which specifies a condition. The appliance performs RNAT processing on those packets that match the condition.

Stateless NAT46 Translation—Stateless NAT46 enables communication between IPv4 and IPv6 networks, by way of IPv4 to IPv6 packet translation and vice versa, without maintaining any session information on the NetScaler appliance. A stateless NAT46 configuration involves an IPv4-IPv6 INAT rule and an NAT46 IPv6 prefix.

Stateful NAT64 Translation—The stateful NAT64 feature enables communication between IPv4 clients and IPv6 servers through IPv6 to IPv4 packet translation, and vice versa, while maintaining session information on the NetScaler appliance. A stateful NAT64 configuration involves an NAT64 rule and an NAT64 IPv6 prefix.

For more information, see "Configuring Network Address Translation."

Multipath TCP Support

NetScaler appliances support Multipath TCP (MPTCP). MPTCP is a TCP/IP protocol extension that identifies and uses multiple paths available between hosts to maintain the TCP session. You must enable MPTCP on a TCP profile and bind it to a virtual server. When MPTCP is enabled, the virtual server functions as an MPTCP gateway and converts MPTCP connections with the clients to TCP connections that it maintains with the servers.
For more information, see "MPTCP (Multi-Path TCP)."

Content Switching

Determines the server to which to send the request on the basis of configured content switching policies. Policy rules can be based on the IP address, URL, and HTTP headers. This allows switching decisions to be based on user and device characteristics such as who the user is, what type of agent is being used, and what content the user requested.
For more information, see "Content Switching."

Global Server Load Balancing (GSLB)

Extends the traffic management capabilities of a NetScaler to include distributed Internet sites and global enterprises. Whether installations are spread across multiple network locations or multiple clusters in a single location, the NetScaler maintains availability and distributes traffic across them. It makes intelligent DNS decisions to prevent users from being sent to a site that is down or overloaded. When the proximity-based GSLB method is enabled, the NetScaler can make load balancing decisions based on the proximity of the client's local DNS server (LDNS) in relation to different sites. The main benefit of the proximity-based GSLB method is faster response time resulting from the selection of the closest available site.
For more information, see "Global Server Load Balancing."

Dynamic Routing

Enables routers to obtain topology information, routes, and IP addresses from neighboring routers automatically. When dynamic routing is enabled, the corresponding routing process listens to route updates and advertises routes. The routing processes can also be placed in passive mode. Routing protocols enable an upstream router to load balance traffic to identical virtual servers hosted on two standalone NetScaler units using the Equal Cost Multipath technique.
For more information, see "Configuring Dynamic Routes."

Link Load Balancing

Load balances multiple WAN links and provides link failover, further optimizing network performance and ensuring business continuity. Ensures that network connections remain highly available, by applying intelligent traffic control and health checks to distribute traffic efficiently across upstream routers. Identifies the best WAN link to route both incoming and outbound traffic based on policies and network conditions, and protects applications against WAN or Internet link failure by providing rapid fault detection and failover.
For more information, see " Link Load Balancing."

TCP Optimization

You can use TCP profiles to optimize TCP traffic. TCP profiles define the way that NetScaler virtual servers process TCP traffic. Administrators can use the built-in TCP profiles or configure custom profiles. After defining a TCP profile, you can bind it to a single virtual server or to multiple virtual servers.
Some of the key optimization features that can be enabled by TCP profiles are:

- TCP keep-alive—Checks the operational status of the peers at specified time intervals to prevent the link from being broken.
- Selective Acknowledgment (SACK)— Improves the performance of data transmission, especially in long fat networks (LFNs).
- TCP window scaling— Allows efficient transfer of data over long fat networks (LFNs).

For more information on TCP Profiles, see "Configuring TCP Profiles."

Web Interface on NetScaler

Provides access to XenApp and XenDesktop resources, which include applications, content, and desktops. Users access resources through a standard Web browser or by using the Citrix XenApp plug-in. The Web Interface runs as a service on port 8080 on the NetScaler appliance. To create Web Interface sites, Java is executed on Apache Tomcat Web server version 6.0.26 on the NetScaler appliance.
Note: Web Interface is supported only on NetScaler nCore releases.
For more information, see "Web Interface."

CloudBridge Connector

The Citrix NetScaler CloudBridge Connector feature, a fundamental part of the Citrix OpenCloud framework, is a tool used to build a cloud-extended data center. The OpenCloud Bridge enables you to connect one or more NetScaler appliances or NetScaler virtual appliances on the cloud-to your network without reconfiguring your network. Cloud hosted applications appear as though they are running on one contiguous enterprise network. The primary purpose of the OpenCloud Bridge is to enable companies to move their applications to the cloud while reducing costs and the risk of application failure. In addition, the OpenCloud Bridge increases network security in cloud environments. An OpenCloud Bridge is a Layer-2 network bridge that connects a NetScaler appliance or NetScaler virtual appliance on a cloud instance to a NetScaler appliance or NetScaler virtual appliance on your LAN. The connection is made through a tunnel that uses the Generic Routing Encapsulation (GRE) protocol. The GRE protocol provides a mechanism for encapsulating packets from a wide variety of network protocols to be forwarded over another protocol. Then Internet Protocol security (IPsec) protocol suite is used to secure the communication between the peers in the OpenCloud Bridge.

For more information, see "CloudBridge."

DataStream

The NetScaler DataStream feature provides an intelligent mechanism for request switching at the database layer by distributing requests on the basis of the SQL query being sent.

When deployed in front of database servers, a NetScaler ensures optimal distribution of traffic from the application servers and Web servers. Administrators can segment traffic according to information in the SQL query and on the basis of database names, user names, character sets, and packet size.

You can configure load balancing to switch requests according to load balancing algorithms, or you can elaborate the switching criteria by configuring content switching to make a decision based on SQL query parameters, such as user name, database names, and command parameters. You can further configure monitors to track the states of database servers.

The advanced policy infrastructure on the NetScaler appliance includes expressions that you can use to evaluate and process the requests. The advanced expressions evaluate traffic associated with MySQL database servers. You can use request-based expressions (expressions that begin with MYSQL.CLIENT and MYSQL.REQ) in advanced policies to make request switching decisions at the content switching virtual server bind point and response-based expressions (expressions that begin with MYSQL.RES) to evaluate server responses to user-configured health monitors.

Note: DataStream is supported for MySQL and MS SQL databases.
For more information, see "DataStream."

Application Acceleration Features

Sep 06, 2013

AppCompress

Uses the gzip compression protocol to provide transparent compression for HTML and text files. The typical 4:1 compression ratio yields up to 50% reduction in bandwidth requirements out of the data center. It also results in significantly improved end-user response time, because it reduces the amount of data that must be delivered to the user's browser.

For more information, see "Compression."

Cache Redirection

Manages the flow of traffic to a reverse proxy, transparent proxy, or forward proxy cache farm. Inspects all requests, and identifies non-cacheable requests and sends them directly to the origin servers over persistent connections. By intelligently redirecting non-cacheable requests back to the origin web servers, the NetScaler appliance frees cache resources and increases cache hit rates while reducing overall bandwidth consumption and response delays for these requests.

For more information, see "Cache Redirection."

AppCache

Helps optimize web content and application data delivery by providing a fast in-memory HTTP/1.1 and HTTP/1.0 compliant web caching for both static and dynamic content. This on-board cache stores the results of incoming application requests even when an incoming request is secured or the data compressed, and then reuses the data to fulfill subsequent requests for the same information. By serving data directly from the on-board cache, the appliance can reduce page regeneration times by eliminating the need to funnel static and dynamic content requests to the server.

For more information, see "Integrated Caching."

TCP Buffering

Buffers the server's response and delivers it to the client at the client's speed, thus offloading the server faster and thereby improving the performance of web sites.

For more information, see "TCP Buffering."

Application Security and Firewall Features

Oct 30, 2013

Denial of Service Attack (DoS) Defense

Detects and stops malicious distributed denial-of-service (DDoS) attacks and other types of malicious attacks before they reach your servers, preventing them from affecting network and application performance. The NetScaler appliance identifies legitimate clients and elevates their priority, leaving suspect clients unable to consume a disproportionate percentage of resources and cripple your site. The appliance provides application-level protection from the following types of malicious attacks:

- SYN flood attacks
- Pipeline attacks
- Teardrop attacks
- Land attacks
- Fraggle attacks
- Zombie connection attacks

The appliance aggressively defends against these types of attacks by preventing the allocation of server resources for these connections. This insulates servers from the overwhelming flood of packets associated with these events.

The appliance also protects network resources from ICMP based attacks by using ICMP rate limiting and aggressive ICMP packet inspection. It performs strong IP reassembly, drops a variety of suspicious and malformed packets, and applies Access Control Lists (ACLs) to site traffic for further protection.

For more information, see "HTTP Denial-of-Service Protection."

Content Filtering

Provides protection from malicious attacks for web sites at the Layer 7 level. The appliance inspects each incoming request according to user-configured rules based on HTTP headers, and performs the action the user configured. Actions can include resetting the connection, dropping the request, or sending an error message to the user's browser. This allows the appliance to screen unwanted requests and reduces your servers' exposure to attacks.
This feature can also analyze HTTP GET and POST requests and filter out known bad signatures, allowing it to defend your servers against HTTP-based attacks.

For more information, see "Content Filtering."

Responder

Functions like an advanced filter and can be used to generate responses from the appliance to the client. Some common uses of this feature are generation of redirect responses, user defined responses, and resets.
For more information, see "Responder."

Rewrite

Modifies HTTP headers and body text. You can use the rewrite feature to add HTTP headers to an HTTP request or response, make modifications to individual HTTP headers, or delete HTTP headers. It also enables you to modify the HTTP body in requests and responses.
When the appliance receives a request or sends a response, it checks for rewrite rules, and if applicable rules exist, it applies them to the request or response before passing it on to the web server or client computer.

For more information, see "Rewrite."

Priority Queuing

Prioritizes user requests to ensure that the most important traffic is serviced first during surges in request volume. You can establish priority based on request URLs, cookies, or a variety of other factors. The appliance places requests in a three-tier queue based on their configured priority, enabling business-critical transactions to flow smoothly even during surges or site attacks.
For more information, see "Priority Queuing."

Surge Protection

Regulates the flow of user requests to servers and controls the number of users that can simultaneously access the resources on the servers, queuing any additional requests once your servers have reached their capacity. By controlling the rate at which connections can be established, the appliance blocks surges in requests from being passed on to your servers, thus preventing site overload.
For more information, see "Surge Protection."

NetScaler Gateway

NetScaler Gateway is a secure application access solution that provides administrators granular application-level policy and action controls to secure access to applications and data while allowing users to work from anywhere. It gives IT administrators a single point of control and tools to help ensure compliance with regulations and the highest levels of information security across and outside the enterprise. At the same time, it empowers users with a single point of access—optimized for roles, devices, and networks—to the enterprise applications and data they need. This unique combination of capabilities helps maximize the productivity of today's mobile workforce.

For more information, see "NetScaler Gateway."

Application Firewall

Protects applications from misuse by hackers and malware, such as cross site scripting attacks, buffer overflow attacks, SQL injection attacks, and forceful browsing, by filtering traffic between each protected web server and users that connect to any web site on that web server. The application firewall examines all traffic for evidence of attacks on web server security or misuse of web server resources, and takes the appropriate action to prevent these attacks from succeeding.
For more information, see "Application Firewall."

Application Visibility Feature

Sep 04, 2013

NetScaler Insight Center

NetScaler Insight Center is a high performance collector that provides end-to-end user experience visibility across Web and HDX (ICA) traffic. It collects HTTP and ICA AppFlow records generated by NetScaler ADC appliances and populates analytical reports covering Layer 3 to Layer 7 statistics. NetScaler Insight Center provides in-depth analysis for the last five minutes of real-time data, and for historical data collected for the last one hour, one day, one week, and one month.
HDX (ICA) analytic dashboard enables you to drill down from HDX Users, Applications, Desktops, and even from gateway-level information. Similarly, HTTP analytics provide a bird's eye view of Web Applications, URLs Accessed, Client IP Addresses and Server IP Addresses, and other dashboards. The administrator can drill down and identify the pain points from any of these dashboards, as appropriate for the use case.

EdgeSight for NetScaler

Support for application performance monitoring based on end user experience. This solution leverages the HTML injection feature to obtain various time values, which are used by EdgeSight server for analysis and report generation. EdgeSight for NetScaler provides a way to monitor the performance benefits of a NetScaler and determine potential bottlenecks in a network.
For more information, see "EdgeSight Monitoring for NetScaler."

Enhanced Application Visibility Using AppFlow

The Citrix NetScaler appliance is a central point of control for all application traffic in the data center. It collects flow and user-session level information valuable for application performance monitoring, analytics, and business intelligence applications. AppFlow transmits this information by using the Internet Protocol Flow Information eXport (IPFIX) format, which is an open Internet Engineering Task Force (IETF) standard defined in RFC 5101. IPFIX (the standardized version of Cisco's NetFlow) is widely used to monitor network flow information. AppFlow defines new Information Elements to represent application-level information.
Using UDP as the transport protocol, AppFlow transmits the collected data, called *flow records*, to one or more IPv4 collectors. The collectors aggregate the flow records and generate real-time or historical reports.

AppFlow provides visibility at the transaction level for HTTP, SSL, TCP, and SSL_TCP flows. You can sample and filter the flow types that you want to monitor.

To limit the types of flows to monitor, by sampling and filtering the application traffic, you can enable AppFlow for a virtual server. AppFlow can also provide statistics for the virtual server.

You can also enable AppFlow for a specific service, representing an application server, and monitor the traffic to that application server.

For more information, see "AppFlow."

Stream Analytics

The performance of your web site or application depends on how well you optimize the delivery of the most frequently requested content. Techniques such as caching and compression help accelerate the delivery of services to clients, but you need to be able to identify the resources that are requested most frequently, and then cache or compress those resources. You can identify the most frequently used resources by aggregating real-time statistics about website or application traffic. Statistics such as how frequently a resource is accessed relative to other resources and how much bandwidth is consumed

by those resources help you determine whether those resources need to be cached or compressed to improve server performance and network utilization. Statistics such as response times and the number of concurrent connections to the application help you determine whether you must enhance server-side resources.

If the web site or application does not change frequently, you can use products that collect statistical data, and then manually analyze the statistics and optimize the delivery of content. However, if you do not want to perform manual optimizations, or if your web site or application is dynamic in nature, you need infrastructure that can not only collect statistical data but can also automatically optimize the delivery of resources on the basis of the statistics. On the NetScaler appliance, this functionality is provided by the Stream Analytics feature. The feature operates on a single NetScaler appliance and collects run-time statistics on the basis of criteria that you define. When used with NetScaler policies, the feature also provides you with the infrastructure that you need for automatic, real-time traffic optimization.

For more information, see "Stream Analytics."

Cloud Integration Feature

Aug 23, 2013

AutoScale

All applications have specific usage patterns that comprise peaks and troughs. These load variations can be dynamic in nature and difficult to predict, given that they depend on several factors that are intrinsic to the use case. Cloud users have to constantly monitor the load on their application fleet and make sure that these variations have minimum impact on end users. During periods of peak usage, when the application fleet is overloaded and end users experience significant latency, they have to deploy additional application instances. During trough periods, the expanded fleet is underutilized. So they might have to remove additional instances or bear unnecessary cost overheads. In most cases, they have to perform these tasks manually.

If your organization uses Citrix CloudPlatform to deploy and manage the cloud environment, users can use the *AutoScale* feature in CloudPlatform, in conjunction with a Citrix NetScaler appliance, to automatically scale their applications as needed. The AutoScale feature is part of the elastic load balancing feature in CloudPlatform. A CloudPlatform user can use the AutoScale feature to specify thresholds for various conditions for automatically scaling the application fleet upward and downward. CloudPlatform, in turn, configures the NetScaler appliance (by using the NetScaler NITRO API) to load balance traffic to the application virtual machines (VMs), monitor application thresholds and performance, and trigger scale-up and scale-down actions to add or remove VMs to or from the application fleet.

As the NetScaler administrator, you do not have to perform any tasks for configuring AutoScale on the NetScaler appliance. However, you might have to be aware of certain prerequisites, and you might have to troubleshoot the configuration if issues arise in the AutoScale configuration. To troubleshoot the configuration, you have to be aware of how CloudPlatform works and what configuration CloudPlatform pushes to the NetScaler appliance. You also need a working knowledge of how to troubleshoot issues on a NetScaler appliance.

For more information about AutoScale, see "AutoScale: Automatic Scaling in the Citrix CloudPlatform Environment."

Getting Started with Citrix NetScaler VPX

Aug 13, 2014

Intended for system and network administrators who install and configure complex networking equipment, this section of the library describes initial setup and basic configuration of the NetScaler virtual appliance product, including the following topics.

Understanding the NetScaler

Updated: 2013-10-28

The Citrix NetScaler product is an application switch that performs application-specific traffic analysis to intelligently distribute, optimize, and secure Layer 4-Layer 7 (L4–L7) network traffic for web applications. For example, a NetScaler bases load balancing decisions on individual HTTP requests instead of on long-lived TCP connections, so that the failure or slowdown of a server is managed much more quickly and with less disruption to clients. The NetScaler feature set can be broadly categorized as consisting of switching features, security and protection features, and server-farm optimization features.

Switching Features

Updated: 2013-09-06

When deployed in front of application servers, a NetScaler ensures optimal distribution of traffic by the way in which it directs client requests. Administrators can segment application traffic according to information in the body of an HTTP or TCP request, and on the basis of L4–L7 header information such as URL, application data type, or cookie. Numerous load balancing algorithms and extensive server health checks improve application availability by ensuring that client requests are directed to the appropriate servers.

Security and Protection Features

Updated: 2013-10-28

NetScaler security and protection features protect web applications from Application Layer attacks. A NetScaler allows legitimate client requests and can block malicious requests. It provides built-in defenses against denial-of-service (DoS) attacks and supports features that protect against legitimate surges in application traffic that would otherwise overwhelm the servers. An available built-in firewall protects web applications from Application Layer attacks, including buffer overflow exploits, SQL injection attempts, cross-site scripting attacks, and more. In addition, the firewall provides identity theft protection by securing confidential corporate information and sensitive customer data.

Optimization Features

Updated: 2013-09-06

Optimization features offload resource-intensive operations, such as Secure Sockets Layer (SSL) processing, data compression, client keep-alive, TCP buffering, and the caching of static and dynamic content from servers. This improves the performance of the servers in the server farm and therefore speeds up applications. A NetScaler supports several transparent TCP optimizations, which mitigate problems caused by high latency and congested network links, accelerating the delivery of applications while requiring no configuration changes to clients or servers.

Understanding Policies and Expressions

A policy defines specific details of traffic filtering and management on a NetScaler. It consists of two parts: the expression and the action. The expression defines the types of requests that the policy matches. The action tells the NetScaler what to do when a request matches the expression. As an example, the expression might be to match a specific URL pattern to a type of security attack, with the action being to drop or reset the connection. Each policy has a priority, and the priorities determine the order in which the policies are evaluated.

When a NetScaler receives traffic, the appropriate policy list determines how to process the traffic. Each policy on the list contains one or more expressions, which together define the criteria that a connection must meet to match the policy.

For all policy types except Rewrite policies, a NetScaler implements only the first policy that a request matches, not any additional policies that it might also match. For Rewrite policies, the NetScaler evaluates the policies in order and, in the case of multiple matches, performs the associated actions in that order. Policy priority is important for getting the results you want.

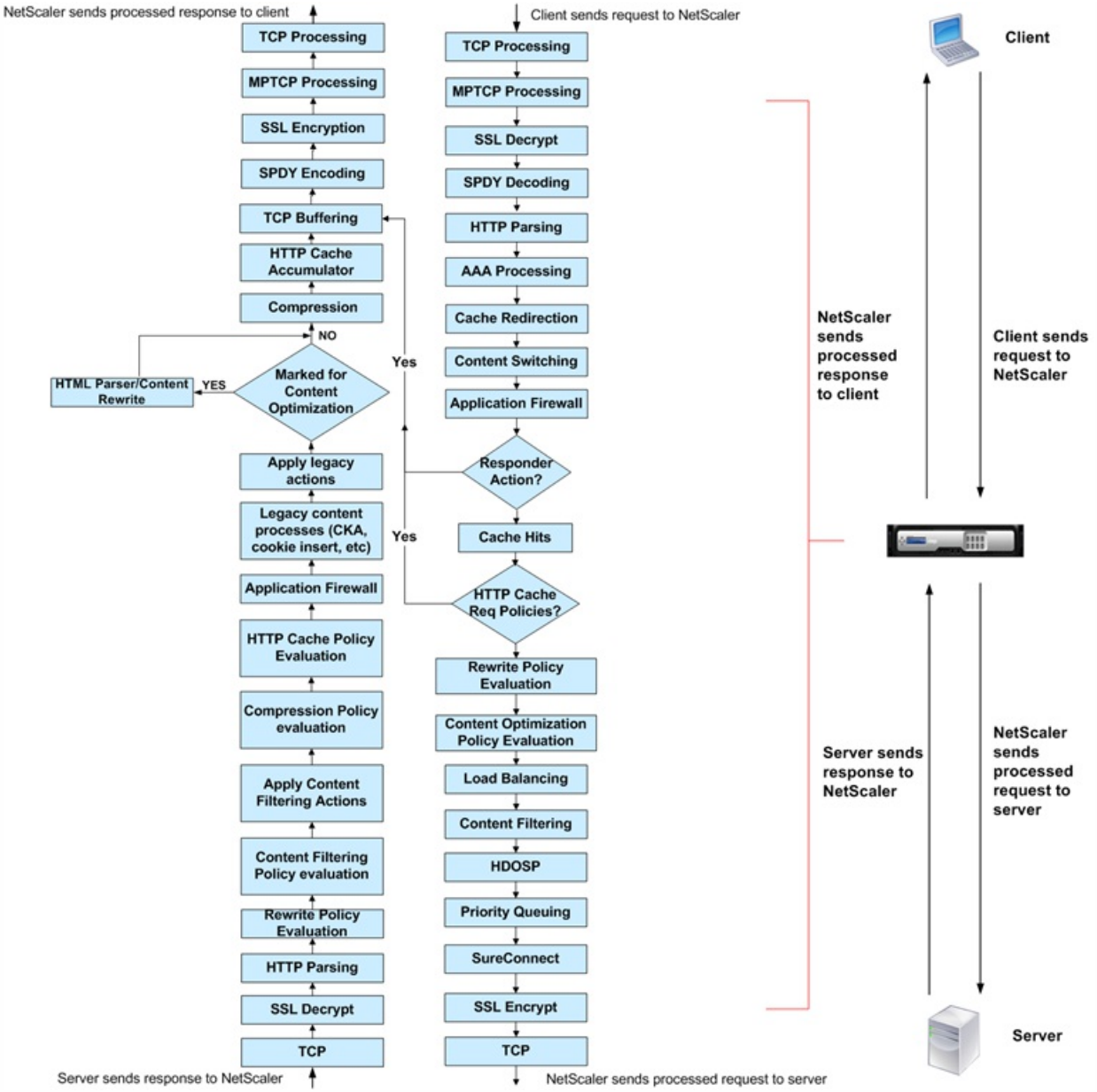
Processing Order of Features

Updated: 2013-08-22

Depending on requirements, you can choose to configure multiple features. For example, you might choose to configure both compression and SSL offload. As a result, an outgoing packet might be compressed and then encrypted before being sent to the client.

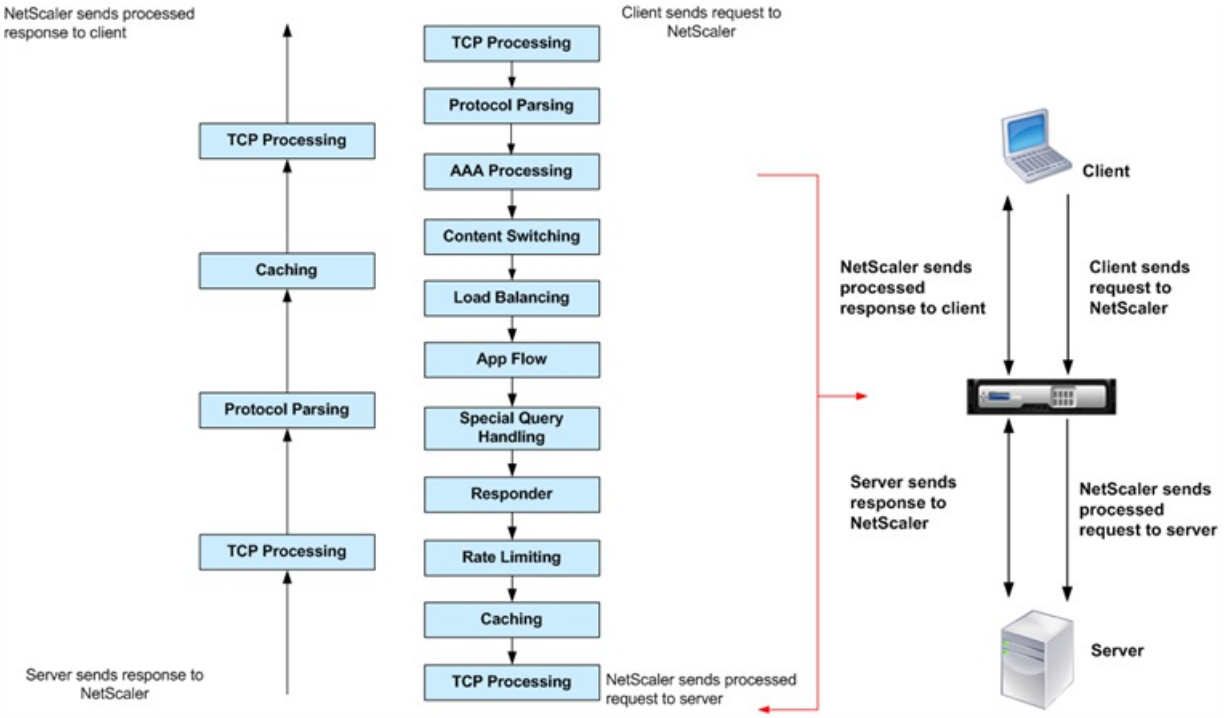
The following figure shows the L7 packet flow in the NetScaler.

Figure 1. L7 Packet Flow Diagram



The following figure shows the DataStream packet flow in the NetScaler. DataStream is supported for MySQL and MS SQL databases. For information about the DataStream feature, see "DataStream."

Figure 2. DataStream Packet Flow Diagram



- Citrix NetScaler Virtual Appliance Overview
- Where Does a NetScaler Appliance Fit in the Network?
- How a NetScaler Communicates with Clients and Servers
- Installing NetScaler Virtual Appliances on XenServer
- Installing NetScaler Virtual Appliances on VMware ESX
- Installing Citrix NetScaler Virtual Appliances on Microsoft Hyper-V Servers
- Installing NetScaler Virtual Appliances on Linux-KVM Platform
- Installing NetScaler VPX on AWS
- Deploying Citrix NetScaler VPX on Microsoft Azure
- Configuring the Basic System Settings
- Understanding Common Network Topologies
- Configuring System Management Settings
- Load Balancing Traffic on a NetScaler Appliance
- Accelerating Load Balanced Traffic by Using Compression
- Securing Load Balanced Traffic by Using SSL
- Setting Up vPath on NetScaler VPX
- Features at a Glance

Citrix NetScaler Virtual Appliance Overview

Oct 06, 2016

The NetScaler virtual appliance product is a virtual NetScaler appliance that can be hosted on Citrix XenServer®, VMware ESX or ESXi, Linux-KVM, and Microsoft Hyper-V virtualization platforms.

A NetScaler virtual appliance supports all the features of a physical NetScaler, except virtual MAC (vMAC) addresses, Layer 2 (L2) mode, and link aggregation control protocol (LACP). VLAN tagging is supported on the NetScaler virtual appliances hosted on the XenServer and on VMware ESX platforms.

For the VLAN tagging feature to work, do one of the following:

- On the Citrix XenServer, configure tagged VLANs on a port on the switch but do NOT configure any VLANs on the XenServer interface attached to that port. The VLAN tags are passed through to the virtual appliance and you can use the tagged VLAN configuration on the virtual appliance.
- On the VMware ESX, set the port group's VLAN ID to 4095 on the VSwitch of VMware ESX server. For more information about setting a VLAN ID on the VSwitch of VMware ESX server, see http://www.vmware.com/pdf/esx3_vlan_wp.pdf.

This overview covers only aspects that are unique to NetScaler virtual appliance.

Note: The terms *NetScaler*, *NetScaler appliance*, and *appliance* are used interchangeably with *NetScaler virtual appliance* unless stated otherwise.

NetScaler Virtual Appliance Setup for the XenServer Platform

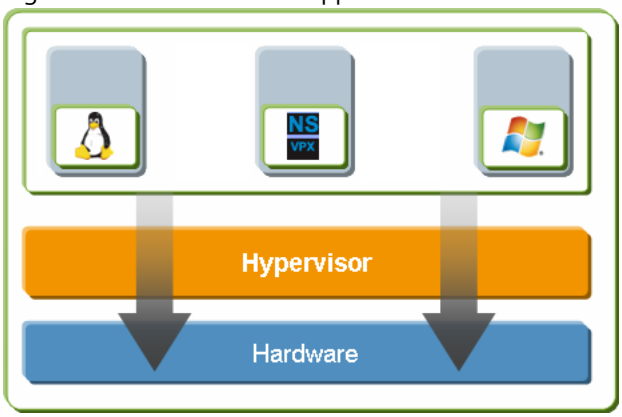
When you set up NetScaler virtual appliance on XenServer, you must use the XenCenter client to install the first NetScaler virtual appliance. Subsequent virtual appliances can be added by using either the XenCenter client or Citrix Command Center.

XenServer

The XenServer® product is a server virtualization platform that offers near bare-metal virtualization performance for virtualized server and client operating systems. XenServer uses the Xen® hypervisor to virtualize each server on which it is installed, enabling each server to host multiple virtual machines simultaneously.

The following figure shows the bare-metal solution architecture of NetScaler virtual appliance on XenServer.

Figure 1. NetScaler Virtual Appliance on XenServer



The bare-metal solution architecture has the following components:

Hardware or physical layer:

Physical hardware components including memory, CPU, network cards, and disk drives.

Xen hypervisor:

Thin layer of software that runs on top of the hardware. The Xen hypervisor gives each virtual machine a dedicated view of the hardware.

Virtual machine:

Operating system hosted on the hypervisor and appearing to the user as a separate physical computer. However, the machine shares physical resources with other virtual machines, and it is portable because the virtual machine is abstracted from physical hardware.

A NetScaler virtual machine, or *virtual appliance*, is installed on the Xen hypervisor and uses paravirtualized drivers to access storage and network resources. It appears to the users as an independent NetScaler appliance with its own network identity, user authorization and authentication capabilities, configuration, applications, and data. The paravirtualization technique enables the virtual machines and the hypervisor to work together to achieve high performance for I/O and for CPU and memory virtualization.

For more information about XenServer, see the XenServer documentation at http://support.citrix.com/product/xens/.

XenCenter

XenCenter® is a graphical virtualization-management interface for XenServer®, enabling you to manage servers, resource pools, and shared storage, and to deploy, manage, and monitor virtual machines from your Windows desktop machine.

Use XenCenter to install NetScaler virtual appliance on XenServer.

For more information about XenCenter, see the XenServer documentation at http://support.citrix.com/product/xens/.

Command Center

Command Center is a management and monitoring solution for Citrix application networking products that include NetScaler, NetScaler virtual appliance, NetScaler Gateway Enterprise Edition, Citrix® Branch Repeater™, Branch Repeater VPX™, and Citrix Repeater™. Command Center enables network administrators and operations teams to manage, monitor, and troubleshoot the entire global application delivery infrastructure from a single, unified console.

This centralized management solution simplifies operations by providing administrators with enterprise-wide visibility and automating management tasks that need to be executed across multiple devices.

Command Center is available with Citrix NetScaler Enterprise and Platinum editions.

You can use Command Center to provision NetScaler virtual appliance on XenServer, and then you can manage and monitor the virtual appliances from Command Center.

Note: You must use the XenCenter client to manage XenServer. You cannot manage XenServer from Command Center.

NetScaler Virtual Appliance Setup for the VMware ESX Platform

The NetScaler virtual appliance setup for the VMware ESX platform requires a VMware ESX or ESXi server and the vSphere client.

VMware ESX and ESXi are virtualization products based on bare-metal architecture, offered by VMware, Inc. Citrix NetScaler virtual appliance can be hosted on a VMware ESX or ESXi server.

For more information about VMware ESX, see http://www.vmware.com/.

The vSphere client is a graphical interface for managing virtual machines on VMware ESX servers. You use the vSphere client to allocate resources on the ESX server to virtual appliances installed on the server or to deallocate resources. For example, you can allocate virtual network ports to a virtual appliance.

For more information about VMware vSphere client, see http://www.vmware.com/.

NetScaler Virtual Appliance Setup for the Microsoft Hyper-V Platform

Updated: 2014-08-07

The NetScaler virtual appliance setup for the Microsoft Hyper-V platform requires Windows Server 2008 R2 or 2012 with the Hyper-V role installed. Like all virtualization systems, Hyper-V enables you to create a virtualized computing environment that results in better utilization of your hardware resources.
Hyper-V is a type 1 hypervisor that comes preinstalled with Windows Server 2008 R2 or 2012. It needs to be enabled as a role on the Windows Server.

For more information about Hyper-V, see http://technet.microsoft.com/en-us/library/cc816638(WS.10).aspx.

NetScaler Virtual Appliance Setup for Linux-KVM Platform

Updated: 2014-03-07

The NetScaler® VPX™ is a virtual NetScaler appliance that can be hosted on a kernel based Virtualization Machine(KVM). The host Linux operating system must be installed on suitable hardware by using virtualization tools such as KVM Module and QEMU. NetScaler VPX runs as a virtual appliance on Linux-KVM server. Like all virtualization systems, KVM enables you to create a virtualized computing environment that results in better utilization of your hardware resources.

Hypervisors Supported on a NetScaler Virtual Appliance

Updated: 2014-08-07

The following table lists the details, such as system ID (sysID) and whether support is available for multiple packet engines (multi-PE), for the different hypervisors supported on a NetScaler virtual appliance.

Table 1. Hypervisors Supported on a NetScaler Virtual Appliance

	VPX on XenServer	VPX on VMware ESX	VPX on Microsoft Hyper-V	VPX on generic KVM	VPX on Amazon Web Services
Hypervisor Version	6.0, 6.1, 6.2	4.1, 5.1, 5.5	2008, 2012	Linux - Fedora 16, Fedora 17, RHEL 6.4	N/A
SysID	450000	450010	450020	450070	450040
Multi-PE Supported	Yes	Yes	Yes	Yes	Yes
Clustering Supported	Yes	Yes	Yes	Yes	No

Licenses	**VPX on XenServer** VPX-10, VPX-200, VPX-1000, VPX-3000, VPX-8000	**VPX on VMware ESX** VPX-10, VPX-200, VPX-1000, VPX-3000, VPX-8000	**VPX on Microsoft Hyper-V** VPX-10, VPX-200, VPX-1000, VPX-3000, VPX-8000	**VPX on generic KVM** VPX-10, VPX-200, VPX-1000, VPX-3000, VPX-8000	**VPX on Amazon Web Services** VPX-10, VPX-200, VPX-1000, VPX-BYOL

Where Does a NetScaler Appliance Fit in the Network?

Jun 24, 2013

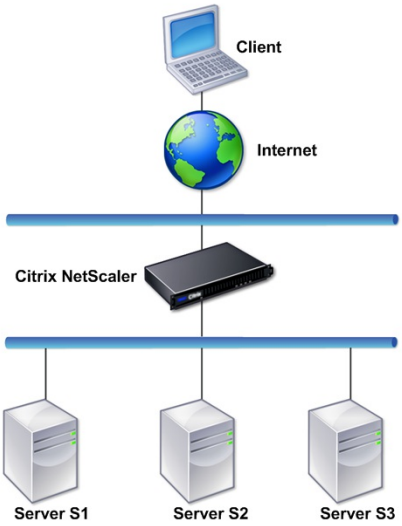
A NetScaler appliance resides between the clients and the servers, so that client requests and server responses pass through it. In a typical installation, virtual servers configured on the appliance provide connection points that clients use to access the applications behind the appliance. In this case, the appliance owns public IP addresses that are associated with its virtual servers, while the real servers are isolated in a private network. It is also possible to operate the appliance in a transparent mode as an L2 bridge or L3 router, or even to combine aspects of these and other modes.

Physical Deployment Modes

Updated: 2013-09-04

A NetScaler appliance logically residing between clients and servers can be deployed in either of two physical modes: inline and one-arm. In inline mode, multiple network interfaces are connected to different Ethernet segments, and the appliance is placed between the clients and the servers. The appliance has a separate network interface to each client network and a separate network interface to each server network. The appliance and the servers can exist on different subnets in this configuration. It is possible for the servers to be in a public network and the clients to directly access the servers through the appliance, with the appliance transparently applying the L4-L7 features. Usually, virtual servers (described later) are configured to provide an abstraction of the real servers. The following figure shows a typical inline deployment.

Figure 1. Inline Deployment



In one-arm mode, only one network interface of the appliance is connected to an Ethernet segment. The appliance in this case does not isolate the client and server sides of the network, but provides access to applications through configured virtual servers. One-arm mode can simplify network changes needed for NetScaler installation in some environments.

For examples of inline (two-arm) and one-arm deployment, see "Understanding Common Network Topologies."

Citrix NetScaler as an L2 Device

Updated: 2013-09-04

A NetScaler functioning as an L2 device is said to operate in L2 mode. In L2 mode, the NetScaler forwards packets between network interfaces when all of the following conditions are met:

- The packets are destined to another device's media access control (MAC) address.
- The destination MAC address is on a different network interface.
- The network interface is a member of the same virtual LAN (VLAN).

By default, all network interfaces are members of a pre-defined VLAN, VLAN 1. Address Resolution Protocol (ARP) requests and responses are forwarded to all network interfaces that are members of the same VLAN. To avoid bridging loops, L2 mode must be disabled if another L2 device is working in parallel with the NetScaler.

For information about how the L2 and L3 modes interact, see "Configuring Modes of Packet Forwarding."

For information about configuring L2 mode, see "Enabling and Disabling Layer 2 Mode."

Citrix NetScaler as a Packet Forwarding Device

Updated: 2014-03-14

A NetScaler appliance can function as a packet forwarding device, and this mode of operation is called L3 mode. With L3 mode enabled, the appliance forwards any received unicast packets that are destined for an IP address that does not belong to the appliance, if there is a route to the destination. The appliance can also route packets between VLANs.

In both modes of operation, L2 and L3, the appliance generally drops packets that are in:

- Multicast frames
- Unknown protocol frames destined for an appliance's MAC address (non-IP and non-ARP)
- Spanning Tree protocol (unless BridgeBPDUs is ON)

For information about how the L2 and L3 modes interact, see "Configuring Modes of Packet Forwarding."

For information about configuring the L3 mode, see "Enabling and Disabling Layer 3 Mode."

How a NetScaler Communicates with Clients and Servers

Aug 30, 2013

A NetScaler appliance is usually deployed in front of a server farm and functions as a transparent TCP proxy between clients and servers, without requiring any client-side configuration. This basic mode of operation is called Request Switching technology and is the core of NetScaler functionality. Request Switching enables an appliance to multiplex and offload the TCP connections, maintain persistent connections, and manage traffic at the request (application layer) level. This is possible because the appliance can separate the HTTP request from the TCP connection on which the request is delivered.

Depending on the configuration, an appliance might process the traffic before forwarding the request to a server. For example, if the client attempts to access a secure application on the server, the appliance might perform the necessary SSL processing before sending traffic to the server.

To facilitate efficient and secure access to server resources, an appliance uses a set of IP addresses collectively known as NetScaler-owned IP addresses. To manage your network traffic, you assign NetScaler-owned IP addresses to virtual entities that become the building blocks of your configuration. For example, to configure load balancing, you create virtual servers to receive client requests and distribute them to services, which are entities representing the applications on your servers.

Understanding NetScaler-Owned IP Addresses

Updated: 2014-03-12

To function as a proxy, a NetScaler appliance uses a variety of IP addresses. The key NetScaler-owned IP addresses are:

NetScaler IP (NSIP) address

The NSIP address is the IP address for management and general system access to the appliance itself, and for communication between appliances in a high availability configuration.

Mapped IP (MIP) address

A MIP address is used for server-side connections. It is not the IP address of the appliance. In most cases, when the appliance receives a packet, it replaces the source IP address with a MIP address before sending the packet to the server. With the servers abstracted from the clients, the appliance manages connections more efficiently.

Virtual server IP (VIP) address

A VIP address is the IP address associated with a virtual server. It is the public IP address to which clients connect. An appliance managing a wide range of traffic may have many VIPs configured.

Subnet IP (SNIP) address

A SNIP address is used in connection management and server monitoring. You can specify multiple SNIP addresses for each subnet. SNIP addresses can be bound to a VLAN.

IP Set

An IP set is a set of IP addresses, which are configured on the appliance as SNIP . An IP set is identified with a meaningful name that helps in identifying the usage of the IP addresses contained in it.

Net Profile

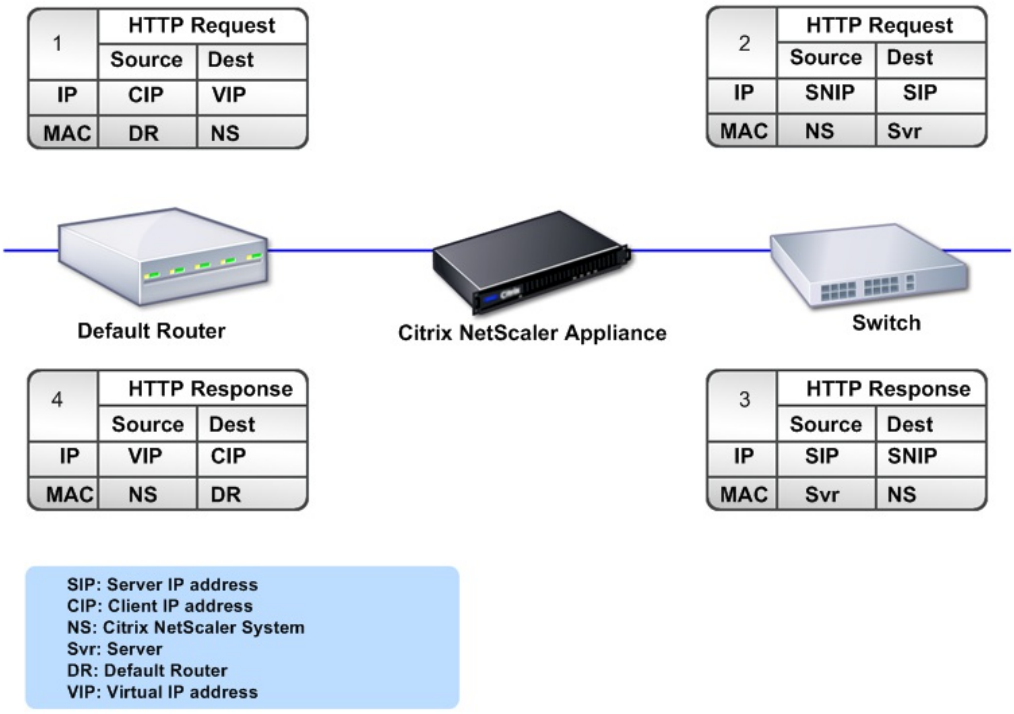
A net profile (or network profile) contains an IP address or an IP set. A net profile can be bound to load balancing or content switching virtual servers, services, service groups, or monitors. During communication with physical servers or peers, the appliance uses the addresses specified in the profile as source IP addresses.

How Traffic Flows Are Managed

Updated: 2014-03-12

Because a NetScaler appliance functions as a TCP proxy, it translates IP addresses before sending packets to a server. When you configure a virtual server, clients connect to a VIP address on the NetScaler instead of directly connecting to a server. As determined by the settings on the virtual server, the appliance selects an appropriate server and sends the client's request to that server. By default, the appliance uses a SNIP address to establish connections with the server, as shown in the following figure.

Figure 1. Virtual Server Based Connections



In the absence of a virtual server, when an appliance receives a request, it transparently forwards the request to the server. This is called the transparent mode of operation. When operating in transparent mode, an appliance translates the source IP addresses of incoming client requests to the SNIP address but does not change the destination IP address. For this mode to work, L2 or L3 mode has to be configured appropriately.

For cases in which the servers need the actual client IP address, the appliance can be configured to modify the HTTP header by inserting the client IP address as an additional field, or configured to use the client IP address instead of a SNIP address for connections to the servers.

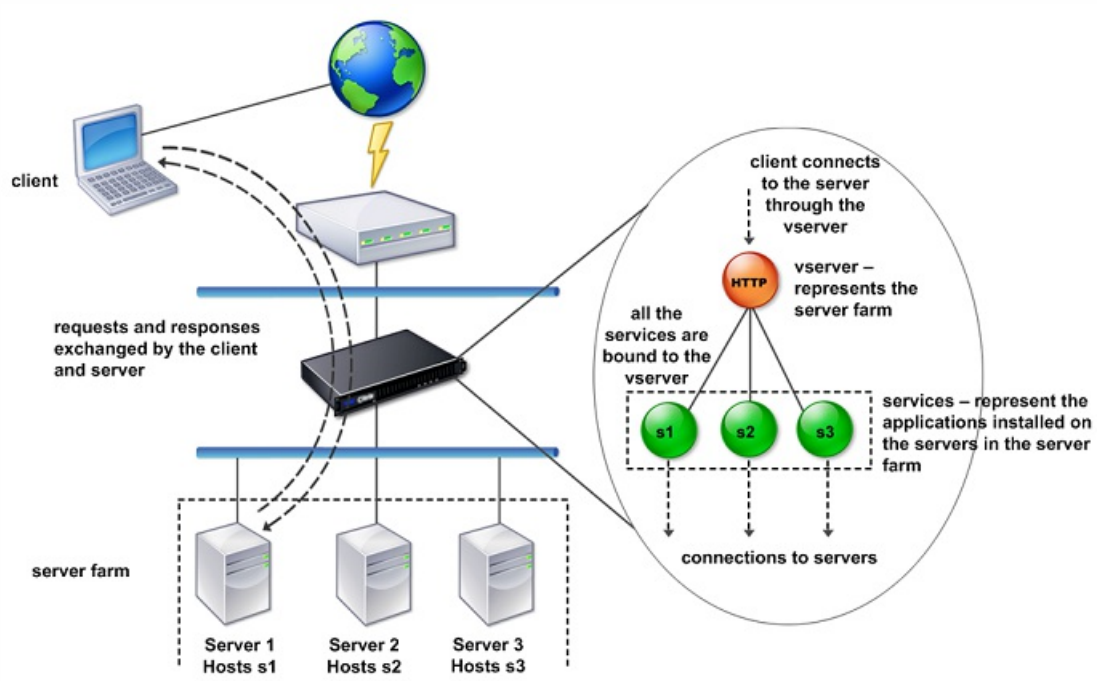
Traffic Management Building Blocks

Updated: 2013-06-24

The configuration of a NetScaler appliance is typically built up with a series of virtual entities that serve as building blocks for traffic management. The building block approach helps separate traffic flows. Virtual entities are abstractions, typically representing IP addresses, ports, and protocol handlers for processing traffic. Clients access applications and resources through these virtual entities. The most commonly used entities are virtual servers and services. Virtual servers represent groups of servers in a server farm or remote network, and services represent specific applications on each server.

Most features and traffic settings are enabled through virtual entities. For example, you can configure an appliance to compress all server responses to a client that is connected to the server farm through a particular virtual server. To configure the appliance for a particular environment, you need to identify the appropriate features and then choose the right mix of virtual entities to deliver them. Most features are delivered through a cascade of virtual entities that are bound to each other. In this case, the virtual entities are like blocks being assembled into the final structure of a delivered application. You can add, remove, modify, bind, enable, and disable the virtual entities to configure the features. The following figure shows the concepts covered in this section.

Figure 2. How Traffic Management Building Blocks Work



A Simple Load Balancing Configuration

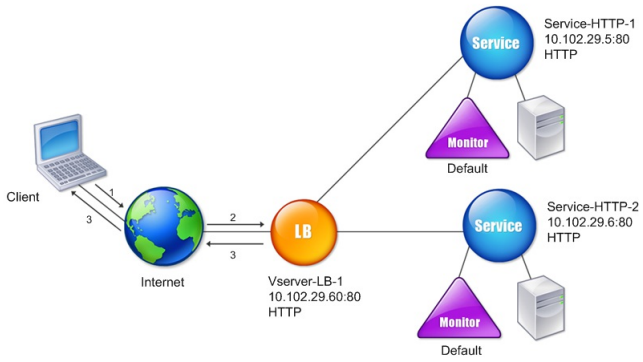
Updated: 2013-08-30

In the example shown in the following figure, the NetScaler appliance is configured to function as a load balancer. For this configuration, you need to configure virtual entities specific to load balancing and bind them in a specific order. As a load balancer, an appliance distributes client requests across several servers and thus optimizes the utilization of resources.

The basic building blocks of a typical load balancing configuration are services and load balancing virtual servers. The services represent the applications on the servers. The virtual servers abstract the servers by providing a single IP address to which the clients connect. To ensure that client requests are sent to a server, you need to bind each service to a virtual server. That is, you must create services for every server and bind the services to a virtual server. Clients use the VIP address to

connect to a NetScaler appliance. When the appliance receives client requests sent to the VIP address, it sends them to a server determined by the load balancing algorithm. Load balancing uses a virtual entity called a monitor to track whether a specific configured service (server plus application) is available to receive requests.

Figure 3. Load Balancing Virtual Server, Services, and Monitors



In addition to configuring the load balancing algorithm, you can configure several parameters that affect the behavior and performance of the load balancing configuration. For example, you can configure the virtual server to maintain persistence based on source IP address. The appliance then directs all requests from any specific IP address to the same server.

Understanding Virtual Servers

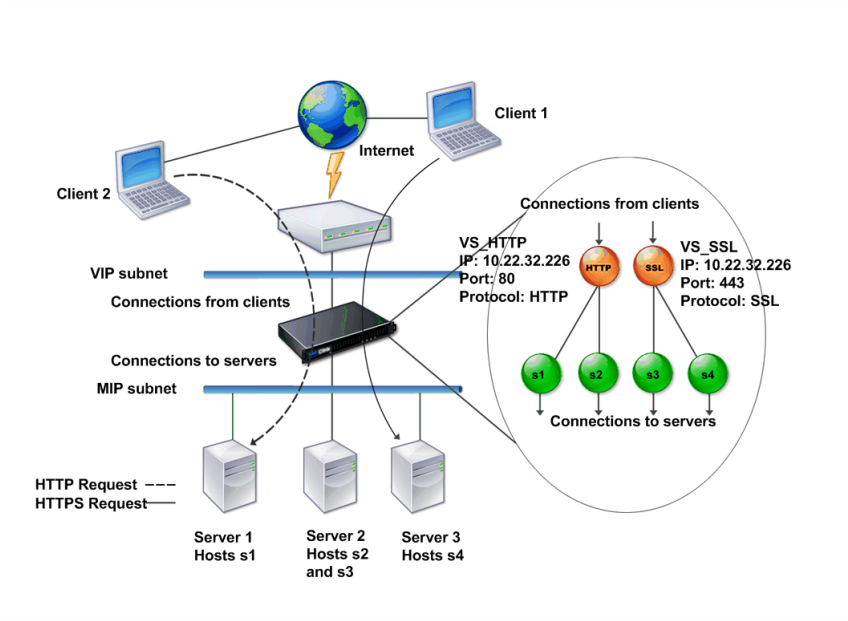
Updated: 2013-09-06

A virtual server is a named NetScaler entity that external clients can use to access applications hosted on the servers. It is represented by an alphanumeric name, virtual IP (VIP) address, port, and protocol. The name of the virtual server is of only local significance and is designed to make the virtual server easier to identify. When a client attempts to access applications on a server, it sends a request to the VIP instead of the IP address of the physical server. When the appliance receives a request at the VIP address, it terminates the connection at the virtual server and uses its own connection with the server on behalf of the client. The port and protocol settings of the virtual server determine the applications that the virtual server represents. For example, a web server can be represented by a virtual server and a service whose port and protocol are set to 80 and HTTP, respectively. Multiple virtual servers can use the same VIP address but different protocols and ports.

Virtual servers are points for delivering features. Most features, like compression, caching, and SSL offload, are normally enabled on a virtual server. When the appliance receives a request at a VIP address, it chooses the appropriate virtual server by the port on which the request was received and its protocol. The appliance then processes the request as appropriate for the features configured on the virtual server.

In most cases, virtual servers work in tandem with services. You can bind multiple services to a virtual server. These services represent the applications running on physical servers in a server farm. After the appliance processes requests received at a VIP address, it forwards them to the servers as determined by the load balancing algorithm configured on the virtual server. The following figure illustrates these concepts.

Figure 4. Multiple Virtual Servers with a Single VIP Address



The preceding figure shows a configuration consisting of two virtual servers with a common VIP address but different ports and protocols. Each of the virtual servers has two services bound to it. The services s1 and s2 are bound to VS_HTTP and represent the HTTP applications on Server 1 and Server 2. The services s3 and s4 are bound to VS_SSL and represent the SSL applications on Server 2 and Server 3 (Server 2 provides both HTTP and SSL applications). When the appliance receives an HTTP request at the VIP address, it processes the request as specified by the settings of VS_HTTP and sends it to either Server 1 or Server 2. Similarly, when the appliance receives an HTTPS request at the VIP address, it processes it as specified by the settings of VS_SSL and it sends it to either Server 2 or Server 3.

Virtual servers are not always represented by specific IP addresses, port numbers, or protocols. They can be represented by wildcards, in which case they are known as wildcard virtual servers. For example, when you configure a virtual server with a wildcard instead of a VIP, but with a specific port number, the appliance intercepts and processes all traffic conforming to that protocol and destined for the predefined port. For virtual servers with wildcards instead of VIPs and port numbers, the appliance intercepts and processes all traffic conforming to the protocol.

Virtual servers can be grouped into the following categories:

Load balancing virtual server

Receives and redirects requests to an appropriate server. Choice of the appropriate server is based on which of the various load balancing methods the user configures.

Cache redirection virtual server

Redirects client requests for dynamic content to origin servers, and requests for static content to cache servers. Cache redirection virtual servers often work in conjunction with load balancing virtual servers.

Content switching virtual server

Directs traffic to a server on the basis of the content that the client has requested. For example, you can create a content switching virtual server that directs all client requests for images to a server that serves images only. Content switching virtual servers often work in conjunction with load balancing virtual servers.

Virtual private network (VPN) virtual server

Decrypts tunneled traffic and sends it to intranet applications.

SSL virtual server

Receives and decrypts SSL traffic, and then redirects to an appropriate server. Choosing the appropriate server is similar to choosing a load balancing virtual server.

Understanding Services

Updated: 2014-03-12

Services represent applications on a server. While services are normally combined with virtual servers, in the absence of a virtual server, a service can still manage application-specific traffic. For example, you can create an HTTP service on a NetScaler appliance to represent a web server application. When the client attempts to access a web site hosted on the web server, the appliance intercepts the HTTP requests and creates a transparent connection with the web server.

In service-only mode, an appliance functions as a proxy. It terminates client connections, uses a SNIP address to establish a connection to the server, and translates the destination IP addresses of incoming client requests to a SNIP address. Although the clients send requests directly to the IP address of the server, the server sees them as coming from the SNIP address. The appliance translates the IP addresses, port numbers, and sequence numbers.

A service is also a point for applying features. Consider the example of SSL acceleration. To use this feature, you must create an SSL service and bind an SSL certificate to the service. When the appliance receives an HTTPS request, it decrypts the traffic and sends it, in clear text, to the server. Only a limited set of features can be configured in the service-only case.

Services use entities called monitors to track the health of applications. Every service has a default monitor, which is based on the service type, bound to it. As specified by the settings configured on the monitor, the appliance sends probes to the application at regular intervals to determine its state. If the probes fail, the appliance marks the service as down. In such cases, the appliance responds to client requests with an appropriate error message or re-routes the request as determined by the configured load balancing policies.

Installing NetScaler Virtual Appliances on XenServer

Aug 23, 2013

To install NetScaler virtual appliances on Citrix XenServer, you must first install XenServer on a machine with adequate system resources. To perform the NetScaler virtual appliance installation, you use Citrix XenCenter, which must be installed on a remote machine that can connect to the XenServer host through the network.

Note: After the initial configuration of the NetScaler appliance, if you want to upgrade the appliance to the latest software release, see Upgrading or Downgrading the System Software.

Prerequisites for Installing NetScaler Virtual Appliances on XenServer

Updated: 2013-08-23

Before you begin installing a virtual appliance, do the following:

- Install XenServer® version 5.6 or later on hardware that meets the minimum requirements.
- Install XenCenter® on a management workstation that meets the minimum system requirements.
- Obtain virtual appliance license files. For more information about virtual appliance licenses, see the *NetScaler VPX Licensing Guide* at http://support.citrix.com/article/ctx122426.

XenServer Hardware Requirements

The following table describes the minimum hardware requirements for a XenServer platform running NetScaler.

Table 1. Minimum System Requirements for XenServer Running NetScaler nCore virtual appliance

Component	Requirement
CPU	2 or more 64-bit x86 CPUs with virtualization assist (Intel-VT or AMD-V) enabled Note: To run NetScaler virtual appliance, hardware support for virtualization must be enabled on the XenServer host. Make sure that the BIOS option for virtualization support is not disabled. Consult your BIOS documentation for more details.
RAM	3 gigabytes (GB)
Disk space	Locally attached storage (PATA, SATA, SCSI) with 40 GB of disk space Note: XenServer installation creates a 4 GB partition for the XenServer host control domain; the remaining space is available for NetScaler virtual appliance and other virtual machines.
Network Interface Card (NIC)	One 1-Gbps NIC Recommended: Two 1-Gbps NICs

For information about installing XenServer, see the XenServer documentation at http://support.citrix.com/product/xens/.

The following table lists the virtual computing resources that XenServer must provide for each NetScaler nCore virtual

appliance .

Table 2. Minimum Virtual Computing Resources Required for Running NetScaler ncore virtual appliance

Component	Requirement
Memory	2 GB
Virtual CPU (VCPU)	2
Virtual network interfaces	2

Note: For production use of NetScaler virtual appliance, Citrix recommends that CPU priority (in virtual machine properties) be set to the highest level, in order to improve scheduling behavior and network latency.

XenCenter System Requirements

XenCenter® is a Windows client application. It cannot run on the same machine as the XenServer® host. The following table describes the minimum system requirements.

Table 3. Minimum System Requirements for XenCenter Installation

Component	Requirement
Operating system	Windows 7, Windows XP, Windows Server 2003, or Windows Vista
.NET framework	Version 2.0 or later
CPU	750 megahertz (MHz) Recommended: 1 gigahertz (GHz) or faster
RAM	1 GB Recommended: 2 GB
Network Interface Card (NIC)	100 megabits per second (Mbps) or faster NIC

For information about installing XenCenter, see the XenServer documentation at http://support.citrix.com/product/xens/.

Installing NetScaler Virtual Appliances on XenServer by Using XenCenter

Updated: 2013-08-23

After you have installed and configured XenServer and XenCenter, you can use XenCenter to install virtual appliances on XenServer. The number of virtual appliances that you can install depends on the amount of memory available on the hardware that is running XenServer.

After you have used XenCenter to install the initial NetScaler virtual appliance (.xva image) on XenServer, you have the option to use Command Center to provision NetScaler virtual appliance. For more information, see the Command Center documentation.

To install NetScaler virtual appliances on XenServer by using XenCenter

1. Start XenCenter on your workstation.
2. On the Server menu, click Add.
3. In the Add New Server dialog box, in the Hostname text box, type the IP address or DNS name of the XenServer that you want to connect to.
4. In the User Name and Password text boxes, type the administrator credentials, and then click Connect. The XenServer name appears in the navigation pane with a green circle, which indicates that the XenServer is connected.
5. In the navigation pane, click the name of the XenServer on which you want to install NetScaler virtual appliance.
6. On the VM menu, click Import.
7. In the Import dialog box, in Import file name, browse to the location at which you saved the NetScaler virtual appliance .xva image file. Make sure that the Exported VM option is selected, and then click Next.
8. Select the XenServer on which you want to install the virtual appliance, and then click Next.
9. Select the local storage repository in which to store the virtual appliance, and then click Import to begin the import process.
10. You can add, modify, or delete virtual network interfaces as required. When finished, click Next.
11. Click Finish to complete the import process.
    Note: To view the status of the import process, click the **Log** tab.
12. If you want to install another virtual appliance, repeat steps 5 through 11.

Installing NetScaler Virtual Appliances on VMware ESX

Aug 07, 2014

Important: You cannot install standard VMware Tools or upgrade the VMware Tools version available on a NetScaler virtual appliance. VMware Tools for a NetScaler virtual appliance are delivered as part of the NetScaler software release.
Before installing NetScaler virtual appliances on VMware ESX, make sure that VMware ESX Server is installed on a machine with adequate system resources. To install NetScaler virtual appliances on VMware ESXi version 4.0 or later, you use VMware vSphere client. The client or tool must be installed on a remote machine that can connect to VMware ESX through the network.

After the installation, you can use vSphere client or vSphere Web Client to manage virtual appliances on VMware ESX 4.0 or later release.

Note:
The VMware vSphere client shows the guest operating system as "Sun Solaris 10" for NetScaler virtual machine. This is by design because VMware ESXi does not recognize FreeBSD.

After the initial configuration of the NetScaler appliance, if you want to upgrade the appliance to the latest software release, see "Upgrading or Downgrading the System Software."

Prerequisites for Installing NetScaler Virtual Appliances on VMware

Updated: 2014-08-07

Before you begin installing a virtual appliance, do the following:

- Install VMware ESX version 4.1 or later on hardware that meets the minimum requirements.
- Install VMware Client on a management workstation that meets the minimum system requirements.
- Download the NetScaler virtual appliance setup files.
- Label the physical network ports of VMware ESX.
- Obtain NetScaler license files. For more information about NetScaler virtual appliance licenses, see the *NetScaler VPX Licensing Guide* at http://support.citrix.com/article/ctx131110.

VMware ESX Hardware Requirements

The following table describes the minimum system requirements for VMware ESX servers running NetScaler ncore virtual appliance.

Table 1. Minimum System Requirements for VMware ESX Servers Running NetScaler nCore virtual appliance

Component	Requirement
CPU	2 or more 64-bit x86 CPUs with virtualization assist (Intel-VT or AMD-V) enabled Note: To run NetScaler virtual appliance, hardware support for virtualization must be enabled on the VMware ESX host. Make sure that the BIOS option for virtualization support is not disabled. For more information, see your BIOS documentation.
RAM	3 GB

Component	Requirement
Disk space	40 GB of disk space available
Network	One 1-Gbps NIC; Two 1-Gbps NICs recommended (The network interfaces must be Intel E1000.)

For information about installing VMware ESX, see http://www.vmware.com/.

The following table lists the virtual computing resources that the VMware ESX server must provide for each NetScaler ncore virtual appliance.

Table 2. Minimum Virtual Computing Resources Required for Running NetScaler ncore virtual appliance

Component	Requirement
Memory	2 GB
Virtual CPU (VCPU)	2 Important: Do not modify the system resources to create a virtual CPU (VCPU) in addition to the two CPUs already allotted to the virtual appliance.
Virtual network interfaces	1 Note: With ESX 4.0 or later, you can install a maximum of 10 virtual network interfaces if the VPX hardware is upgraded version to 7 or higher.
Disk space	20 GB Note: This is in addition to any disk requirements for the hypervisor.

Note: For production use of NetScaler virtual appliance, the full memory allocation must be reserved. CPU cycles (in MHz) equal to at least the speed of one CPU core of the ESX should also be reserved.

VMware vSphere Client System Requirements

VMware vSphere is a client application that can run on Windows and Linux operating systems. It cannot run on the same machine as the VMware ESX server. The following table describes the minimum system requirements.

Table 3. Minimum System Requirements for VMware vSphere Client Installation

Component	Requirement
Operating system	For detailed requirements from VMware, search for the "vSphere Compatibility Matrixes" PDF file at http://kb.vmware.com/.
CPU	750 megahertz (MHz); 1 gigahertz (GHz) or faster recommended
RAM	1 GB; 2 GB recommended

Component	Requirement
Network Interface Card (NIC)	100 Mbps or faster NIC

OVF Tool 1.0 System Requirements

OVF Tool is a client application that can run on Windows and Linux systems. It cannot run on the same machine as the VMware ESX server. The following table describes the minimum system requirements.

Table 4. Minimum System Requirements for OVF Tool Installation

Component	Requirement
Operating system	For detailed requirements from VMware, search for the "OVF Tool User Guide" PDF file at http://kb.vmware.com/.
CPU	750 MHz minimum, 1 GHz or faster recommended.
RAM	1 GB Minimum, 2 GB recommended.
Network Interface Card (NIC)	100 Mbps or faster NIC

For information about installing OVF, search for the "OVF Tool User Guide" PDF file at http://kb.vmware.com/.

Downloading the NetScaler virtual appliance Setup Files

The NetScaler virtual appliance setup package for VMware ESX follows the Open Virtual Machine (OVF) format standard. You can download the files from MyCitrix.com. You need a My Citrix account to log on. If you do not have a My Citrix account, access the home page at http://www.mycitrix.com, click the New Users link, and follow the instructions to create a new My Citrix account.

Once logged on, navigate the following path from the My Citrix home page:

MyCitrix.com > Downloads > NetScaler > Virtual Appliances.

Copy the following files to a workstation on the same network as the ESX server. Copy all three files into the same folder.

- NSVPX-ESX-<release number>-<build number>-disk1.vmdk (for example, NSVPX-ESX-9.3-39.8-disk1.vmdk)
- NSVPX-ESX-<release number>-<build number>.ovf (for example, NSVPX-ESX-9.3-39.8.ovf)
- NSVPX-ESX-<release number>-<build number>.mf (for example, NSVPX-ESX-9.3-39.8.mf)

Labeling the Physical Network Ports of VMware ESX

Before installing a NetScaler virtual appliance, label of all the interfaces that you plan to assign to virtual appliances, in a unique format. Citrix recommends the following format: NS_NIC_1_1, NS_NIC_1_2, and so on. In large deployments, labeling in a unique format helps in quickly identifying the interfaces that are allocated to the NetScaler virtual appliance

among other interfaces used by other virtual machines, such as Windows and Linux. Such labeling is especially important when different types of virtual machines share the same interfaces.

To label the physical network ports of VMware ESX server

1. Log on to the VMware ESX server by using the vSphere client.
2. On the vSphere client, select the Configuration tab, and then click Networking.
3. At the top-right corner, click Add Networking.
4. In the Add Network Wizard, for **Connection Type**, select **Virtual Machine**, and then click Next.
5. Scroll through the list of vSwitch physical adapters, and choose the physical port that will map to interface 1/1 on the virtual appliances.
6. Enter NS_NIC_1_1 as the name of the vSwitch that will be associated with interface 1/1 of the virtual appliances.
7. Click Next to finish the vSwitch creation. Repeat the procedure, beginning with step 2, to add any additional interfaces to be used by your virtual appliances. Label the interfaces sequentially, in the correct format (for example, NS_NIC_1_2).

Installing NetScaler Virtual Appliances on VMware ESX 4.0 or Later

Updated: 2014-05-20

After you have installed and configured VMware ESX 4.0 or later, you can use the VMware vSphere client to install virtual appliances on the VMware ESX. The number of virtual appliances that you can install depends on the amount of memory available on the hardware that is running VMware ESX.

To install NetScaler virtual appliances on VMware ESX 4.0 or later by using VMware vSphere Client

1. Start the VMware vSphere client on your workstation.
2. In the IP address / Name text box, type the IP address of the VMware ESX server that you want to connect to.
3. In the User Name and Password text boxes, type the administrator credentials, and then click Login.
4. On the File menu, click Deploy OVF Template.
5. In the Deploy OVF Template dialog box, in Deploy from file, browse to the location at which you saved the NetScaler virtual appliance setup files, select the .ovf file, and click Next.
6. Map the networks shown in the virtual appliance OVF template to the networks that you configured on the ESX host. Click Next to start installing a virtual appliance on VMware ESX. When installation is complete, a pop-up window informs you of the successful installation.
7. You are now ready to start the NetScaler virtual appliance. In the navigation pane, select the NetScaler virtual appliance that you have just installed and, from the right-click menu, select Power On. Click the Console tab to emulate a console port.
8. If you want to install another virtual appliance, repeat steps 4 through 6.

Installing Citrix NetScaler Virtual Appliances on Microsoft Hyper-V Servers

Feb 09, 2015

Note:

- The NetScaler virtual appliance is supported on Microsoft Hyper-V Server 2008 R2, Microsoft Hyper-V Server 2012 and 2012 R2.
- Intermediate System-to-Intermediate System (ISIS) protocol is not supported on the NetScaler VPX virtual appliance hosted on the HyperV-2012 platform.

To install Citrix NetScaler virtual appliances on Microsoft Windows Server, you must first install Windows Server, with the Hyper-V role enabled, on a machine with adequate system resources. While installing the Hyper-V role, be sure to specify the network interface cards (NICs) on the server that Hyper-V will use to create the virtual networks. You can reserve some NICs for the host. Use Hyper-V Manager to perform the NetScaler virtual appliance installation.
NetScaler virtual appliance for Hyper-V is delivered in virtual hard disk (VHD) format. It includes the default configuration for elements such as CPU, network interfaces, and hard-disk size and format. After you install NetScaler virtual appliance, you can configure the network adapters on virtual appliance, add virtual NICs, and then assign the NetScaler IP address, subnet mask, and gateway, and complete the basic configuration of the virtual appliance.

Note:
After the initial configuration of the NetScaler appliance, if you want to upgrade the appliance to the latest software release, see "Upgrading or Downgrading the System Software."

Prerequisites for Installing NetScaler Virtual Appliance on Microsoft Servers

Updated: 2014-08-07

Before you begin installing a virtual appliance, do the following:

- Enable the Hyper-V role on Windows Servers . For more information, see http://technet.microsoft.com/en-us/library/ee344837(WS.10).aspx.
- Download the virtual appliance setup files.
- Obtain NetScaler virtual appliance license files. For more information about NetScaler virtual appliance licenses, see the *NetScaler VPX Licensing Guide* at http://support.citrix.com/article/ctx131110.

Microsoft Server Hardware Requirements

The following table describes the minimum system requirements for Microsoft Servers .

Table 1. Minimum System Requirements for Microsoft Servers

Component	Requirement
CPU	1.4 GHz 64-bit processor
RAM	3 GB

Component	Requirement
Disk Space	32 GB or greater

The following table lists the virtual computing resources for each NetScaler virtual appliance.

Table 2. Minimum Virtual Computing Resources Required for Running NetScaler Virtual Appliance

Component	Requirement
RAM	2 GB
Virtual CPU	2
Disk Space	20 GB
Virtual Network Interfaces	1

Downloading the NetScaler Virtual Appliance Setup Files

NetScaler virtual appliance for Hyper-V is delivered in virtual hard disk (VHD) format. You can download the files from MyCitrix.com. You will need a My Citrix account to log on. If you do not have a My Citrix account, access the home page at http://www.mycitrix.com, click the New Users link, and follow the instructions to create a new My Citrix account.

To download the NetScaler virtual appliance setup files

1. In a Web browser, go to http://www.citrix.com/ and click My Citrix.
2. Type your user name and password.
3. Click Downloads.
4. In Search Downloads by Product, select NetScaler.
5. Under Virtual Appliances, click NetScaler VPX.
6. Copy the compressed file to your server.

Installing NetScaler Virtual Appliance on Microsoft Servers

Updated: 2014-08-07

After you have enabled the Hyper-V role on Microsoft Server and extracted the virtual appliance files, you can use Hyper-V Manager to install NetScaler virtual appliance. After you import the virtual machine, you need to configure the virtual NICs by associating them to the virtual networks created by Hyper-V.

You can configure a maximum of eight virtual NICs. Even if the physical NIC is DOWN, the virtual appliance assumes that the virtual NIC is UP, because it can still communicate with the other virtual appliances on the same host (server).
Note: You cannot change any settings while the virtual appliance is running. Shut down the virtual appliance and then make changes.

To install NetScaler Virtual Appliance on Microsoft Server by using Hyper-V

Manager

1. To start Hyper-V Manager, click **Start**, point to **Administrative Tools**, and then click **Hyper-V Manager**.
2. In the navigation pane, under Hyper-V Manager, select the server on which you want to install NetScaler virtual appliance.
3. On the **Action** menu, click **Import Virtual Machine**.
4. In the **Import Virtual Machine** dialog box, in **Location**, specify the path of the folder that contains the NetScaler virtual appliance software files, and then select **Copy the virtual machine (create a new unique ID)**. This folder is the parent folder that contains the Snapshots, Virtual Hard Disks, and Virtual Machines folders.
   Note: If you received a compressed file, make sure that you extract the files into a folder before you specify the path to the folder.
5. Click **Import**.
6. Verify that the virtual appliance that you imported is listed under **Virtual Machines**.
7. To install another virtual appliance, repeat steps 2 through 6.
   Important: Make sure that you extract the files to a different folder in step 4.

To configure virtual NICs on the NetScaler Virtual Appliance

1. Select the virtual appliance that you imported, and then on the **Action** menu, select **Settings**.
2. In the **Settings for** <virtual appliance name> dialog box, click **Add Hardware** in the left pane.
3. In the right pane, from the list of devices, select **Network Adapter**.
4. Click **Add**.
5. Verify that **Network Adapter** (**not connected**) appears in the left pane.
6. Select the network adapter in the left pane.
7. In the right pane, from the **Network** drop-down list, select the virtual network to connect the adapter to.
8. To select the virtual network for additional network adapters that you want to use, repeat steps 6 and 7.
9. Click **Apply**, and then click **OK**.

To configure NetScaler Virtual Appliance

1. Right-click the virtual appliance that you previously installed, and then select **Start**.
2. Access the console by double-clicking the virtual appliance.
3. Type the NetScaler IP address, subnet mask, and gateway for your virtual appliance.

You have completed the basic configuration of your virtual appliance. Type the IP address in a Web browser to access the virtual appliance.

Installing NetScaler Virtual Appliances on Linux-KVM Platform

Mar 09, 2015

To set up NetScaler VPX for the Linux-KVM platform, you can use the graphical Virtual Machine Manager (Virt-Manager) application. If you prefer the Linux-KVM command line, you can use the virsh program.

The host Linux operating system must be installed on suitable hardware by using virtualization tools such as KVM Module and QEMU. The number of virtual machines (VMs) that can be deployed on the hypervisor depends on the application requirement and the chosen hardware.

After you provision a NetScaler virtual appliance, you can add additional interfaces.

Limitations and Usage Guidelines

General Recommendations

To avoid unpredictable behavior, apply the following recommendations:

- Do not change the MTU of the vnet interface associated with the NetScaler VM. Shut down the NetScaler VM before modifying any configuration parameters, such as Interface modes or CPU.
- Do not force a shutdown of the NetScaler VM. That is, do not use the **Force off** command.
- Any configurations done on the host Linux might or might not be persistent, depending on your Linux distribution settings. You can choose to make these configurations persistent to ensure consistent behavior across reboots of host Linux operating system.
- The `.raw` file has to be unique for each of the NetScaler VPX instance provisioned.

Limitations

A NetScaler VPX setup on the Linux-KVM platform has the following limitations:

- VLAN tagging is not supported on Netscaler-VPX operating on MacVTap-Bridge, MacVTap-Private, MacVTap-VEPA, or MacVTap-Passthrough interface Mode.
- L2 mode is not supported on Netscaler VPX running on a Linux-KVM host.
- Interface parameter configurations, such as speed, duplex, and auto-negotiation are not supported
- Interface events, such as link UP and DOWN are not supported. Because these events are not reported, the following features are not supported:
  - Static link aggregation
  - Dynamic route advertisement for connected networks
  - Monitored static routes
  - Avoiding split brains in a high availability (HA) setup
  - Partial failure detection in an HA setup
- LACP is not supported on Netscaler VPX operating in Bridge, MacVTap-Bridge, MacVTap-Private, or MacVTap-VEPA interface mode.
- Live Migration of the Netscaler VPX running on KVM is not supported.
- When a VLAN tagged packet destined for a guest VM is received on an Intel IXGBE 10G interface of a KVM host running on Red Hat Enterprise Linux (RHEL) 6.4, the IXGBE driver of this distribution strips the VLAN tag before sending it to the

guest VM (in this case, NetScaler VPX). Because of this host behavior, a NetScaler VPX instance running on RHEL6.4 does not receive the intended VLAN tagged packets.

Prerequisites for Installing NetScaler VPX Virtual Appliances on Linux-KVM Platform

Sep 30, 2015

Hardware Requirements

The following table describes the minimum system requirements for Linux-KVM servers running NetScaler VPX.

Component	Requirement
CPU	• 64-bit x86 processors with the hardware virtualization features included in the AMD-V and Intel VT-X processors. To test whether your CPU supports Linux host, enter the following command at the host Linux shell prompt: **.egrep'^flags.*(vmx|svm)'/proc/cpuinfo** If the BIOS settings for the above extension are disabled, you must enable them in BIOS. • Provide at least 2 CPU cores to Host Linux. • There is no specific recommendation for processor speed, but higher the speed, the better the performance of the VM application.
Memory (RAM)	Minimum 4 GB for the host Linux kernel. Add additional memory as required by the VMs.
Hard Disk	Calculate the space for Host Linux kernel and VM requirements. A single NetScaler VPX VM requires 20 GB of disk space.

Software Requirements

The Host kernel used must be a 64-bit Linux kernel, release 2.6.20 or later, with all virtualization tools. Citrix recommends newer kernels, such as 3.6.11-4 and later.

Many Linux distributions such as Red Hat, Centos, and Fedora, have tested kernel versions and associated virtualization tools.

Guest VM Hardware Requirements

NetScaler VPX supports only IDE hard disk type. The Hard Disk Type has been configured in the XML file which is a part of the NetScaler package.

Networking Requirements

NetScaler VPX supports only virtIO para-virtualized network interfaces.

Source Interface and Modes

The source device type can be either Bridge or MacVTap. In case of MacVTap, four modes are possible - VEPA, Bridge,

Private and Pass-through.
The following tables list the types of interfaces that you can use and the supported traffic types.
For best performance by the NetScaler instance, make sure that the gro and lro capabilities are switched off on the source interfaces

Table 1. Interface Types

Interface Type	Considerations
Source: Bridge	• Linux Bridge. • Ebtables and iptables settings on host Linux might filter the traffic on the bridge if you do not choose the correct setting or disable IPtable services.
Source: MacVTap Mode : VEPA	• Better performance than a bridge. • Interfaces from the same lower device can be shared across the VMs. • Inter-VM communication using the same lower device is possible only if upstream or downstream switch supports VEPA mode.
Source: MacVTap Mode : Private	• Better performance than a bridge. • Interfaces from the same lower device can be shared across the VMs. • Inter-VM communication using the same lower device is not possible.
Source: MacVTap Mode : Bridge	• Better as compared to bridge. • Interfaces out of same lower device can be shared across the VMs. • Inter-VM communication using the same lower device is possible, if lower device link is UP.
Source: MacVTap Mode : Pass-through	• Better as compared to bridge. • Interfaces out of same lower device cannot be shared across the VMs. • Only one VM can use the lower device.

Table 2. Verified Traffic Types

Test Case	Bridge	MacVTap			
		VEPA	Private	Bridge	Pass-through
Untagged IPv4	S	S	S	S	S
Tagged IPv4	S	NS	NS	NS	NS

Test Case	Bridge	MacVTap			
		VEPA	Private	Bridge	Pass-through
IPv4 L3 Forwarding	S	S	S	S	S
IPv4 Endpoint Traffic	S	S	S	S	S
Broadcast/Multicast Traffic	S	S	S	S	S
Untagged IPv6	S	S	S	S	S
Tagged IPv6	S	NS	NS	NS	NS
IPv6 L3 Forwarding	S	S	S	S	S
IPv6 Endpoint Traffic	S	S	S	S	S

S - Supported.

NS - Not Supported.

Properties Of Source Interfaces

Make sure that you switch off the generic-receive-offload (gro) and large-receive-offload (lro) capabilities of the source interfaces. To switch off the gro and lro capabilities, run the following commands at the host Linux shell prompt.

ethtool -K eth6 gro off

ethool -K eth6 lro off

Example

```
[root@localhost ~]# ethtool -K eth6
  Offload parameters for eth6:
    rx-checksumming: on
    tx-checksumming: on
    scatter-gather: on
  tcp-segmentation-offload: on
  udp-fragmentation-offload: off
  generic-segmentation-offload: on
  generic-receive-offload: off
  large-receive-offload: off
  rx-vlan-offload: on
  tx-vlan-offload: on
  ntuple-filters: off
  receive-hashing: on
[root@localhost ~]#
```

Example

If the host Linux bridge is used as a source device, as in the following example, gro and lro capabilities must be switched

off on the vnet interfaces, which are the virtual interfaces connecting the host to the guest VMs.

```
[root@localhost ~]# brctl show eth6_br
bridge name     bridge id       STP enabled     interfaces
eth6_br         8000.00e0ed1861ae       no          eth6
                                        vnet0
                                        vnet2
[root@localhost ~]#
```

In the above example, the two virtual interfaces are derived from the eth6_br and are represented as vnet0 and vnet2. Run the following commands to switch off gro and lro capabilities on these interfaces.

```
ethtool –K vnet0 gro off
 ethtool –K vnet2 gro off
 ethtool –K vnet0 lro off
  ethtool –K vnet2 lro off
```

Module Required

For better network performance, make sure the vhost_net module is present in the Linux host. To check the existence of vhost_net module, run the following command on the Linux host :

```
lsmod | grep "vhost_net"
```

If vhost_net is not yet running, enter the following command to run it:

```
modprobe vhost_net
```

Provisioning the NetScaler Virtual Appliance by using OpenStack

Mar 10, 2015

You can provision a NetScaler vpx instance in an Openstack environment either by using the OpenStack command line interface or the OpenStack dashboard or GUI.

Provisioning a NetScaler instance, optionally involves using data from the config drive. Config drive is a special configuration drive that attaches to the instance when it boots. This configuration drive can be used to pass networking configuration like management IP address, network mask, default gateway etc, which the instance can mount and access before you configure the network settings for the instance.

When OpenStack provisions a NetScaler instance, it first detects that the instance is booting in an OpenStack environment by reading a specific BIOS string that indicates OpenStack. This string is 'OpenStack Foundation' and for Redhat Linux distributions, the string is stored in /etc/nova/release. This is a standard mechanism that is available in all OpenStack implementations based on KVM hyper-visor platform. The drive should have a specific OpenStack label.

If the config drive is detected, the instance attempts to read the following information from the file name specified in the nova boot command. In the steps mentioned below, the file is referred as userdata:

- Management IP address
- Network mask
- Default gateway

Once the parameters are successfully read, they are populated in the NetScaler stack. This helps in managing the instance remotely. If the parameters are not read successfully or the config drive is not available, the instance transitions to the default behavior, which is:

- The instance attempts to retrieve the IP address information from DHCP
- If DHCP fails or times-out, the instance comes up with default network configuration (192.168.100.1/16)

Provisioning the NetScaler Virtual Appliance by using OpenStack Using Command Line Interface

Updated: 2015-02-16

You can provision a NetScaler appliance in an OpenStack environment. Provisioning a NetScaler Virtual Appliance on OpenStack involves the following three steps:

1. Extracting the .raw file from the .ova file
2. Building an OpenStack image from the raw image
3. Provisioning a NetScaler instance

To provision a NetScaler instance in an OpenStack environment, complete the following steps:

1. Extract the .raw file from the .ova file.
   ```
   tar xvzf NetScaler1000V-KVM-10.5-49.3_nc.ova
   NetScaler1000V-KVM.xml
   NetScaler1000V-KVM-10.5-49.3_nc.raw
   checksum.txt
   ```
2. Build an OpenStack image using the .raw file extracted in step 1.
   ```
   glance image-create --name="NS-VPX-10-1-127-1 " --property hw_disk_bus=ide --is-public=true
   ```


--container-format=bare --disk-format=raw < NetScaler1000V-KVM-10.1-127.1_nc.raw

In the above command, NS-VPX-10-1-127-1 is the name of the OpenStack image that you want to create. NetScaler1000V-KVM-10.1-127.1_nc.raw is the raw file that was extracted from the ova file. The raw file is the input for creating the OpenStack image.

The following illustration provides a sample output for the glance image-create command.

```
+--------------------------+--------------------------------------+
| Property                 | Value                                |
+--------------------------+--------------------------------------+
| Property 'hw_disk_bus'   | ide                                  |
| checksum                 | 65b027c72d668abe9941716325731872     |
| container_format         | bare                                 |
| created_at               | 2014-06-12T06:15:50                  |
| deleted                  | False                                |
| deleted_at               | None                                 |
| disk_format              | raw                                  |
| id                       | 102d3621-c6b7-4823-bbf3-2ff47476cdc0 |
| is_public                | True                                 |
| min_disk                 | 0                                    |
| min_ram                  | 0                                    |
| name                     | NS-VPX-10-5-49-3                     |
| owner                    | e6f3a3cc7f764f639370060f1d121a88     |
| protected                | False                                |
| size                     | 21474836480                          |
| status                   | active                               |
| updated_at               | 2014-06-12T06:21:41                  |
| virtual_size             | None                                 |
+--------------------------+--------------------------------------+
```

3. After an OpenStack image is created, provision the NetScaler virtual appliance instance.

   nova boot --image NS-VPX-10-1-127-1 --config-drive=true --user-data ./userdata.txt --flavor m1.medium --nic net-id=b8c5acee-36b7-4517-af0e-80f8729aa82e vpx10_1_u

   In the above command, userdata.txt is the file which contains the details like, IP address, netmask, and default gateway for the NetScaler instance. The userdata file is a user customizable file. vpx10_1_u is the name of the virtual appliance that you want to provision.

   The following illustration gives a sample output of the nova boot command.

```
+--------------------------------------+------------------------------------------------------------+
| Property                             | Value                                                      |
+--------------------------------------+------------------------------------------------------------+
| OS-DCF:diskConfig                    | MANUAL                                                     |
| OS-EXT-AZ:availability_zone          | nova                                                       |
| OS-EXT-STS:power_state               | 0                                                          |
| OS-EXT-STS:task_state                | scheduling                                                 |
| OS-EXT-STS:vm_state                  | building                                                   |
| OS-SRV-USG:launched_at               | -                                                          |
| OS-SRV-USG:terminated_at             | -                                                          |
| accessIPv4                           |                                                            |
| accessIPv6                           |                                                            |
| adminPass                            | mXB4C62mGcxb                                               |
| config_drive                         | True                                                       |
| created                              | 2014-06-12T06:46:24Z                                       |
| flavor                               | m1.medium (3)                                              |
| hostId                               |                                                            |
| id                                   | 6c98e03b-6e8b-4551-a051-0f1d2d88b0c2                       |
| image                                | NS-VPX-10-5-49-3 (102d3621-c6b7-4823-bbf3-2ff47476cdc0)    |
| key_name                             | -                                                          |
| metadata                             | {}                                                         |
| name                                 | NS1000v-10-5                                               |
| os-extended-volumes:volumes_attached | []                                                         |
| progress                             | 0                                                          |
| security_groups                      | default                                                    |
| status                               | BUILD                                                      |
| tenant_id                            | e6f3a3cc7f764f639370060f1d121a88                           |
| updated                              | 2014-06-12T06:46:24Z                                       |
| user_id                              | d3df0f0cabd0422dafcf29c97f96ff1d                           |
+--------------------------------------+------------------------------------------------------------+
[root@redhatkvm tagma49_3]#
```

The following illustration shows a sample of the xml file. The values within the <PropertySection> </PropertySection> tags are the values which is user configurable and holds the information like, IP address, netmask, and default gateway.

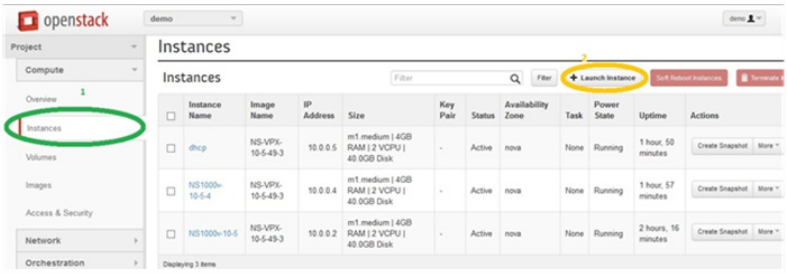
```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Environment xmlns:oe="http://schemas.dmtf.org/ovf/environment/1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
oe:id=""
xmlns="http://schemas.dmtf.org/ovf/environment/1">
<PlatformSection>
<Kind>NOVA</Kind>
<Version>2013.1</Version>
<Vendor>Openstack</Vendor>
<Locale>en</Locale>
</PlatformSection>
<PropertySection>
 <Property oe:key="com.citrix.netscaler.ovf.version" oe:value="1.0"/>
 <Property oe:key="com.citrix.netscaler.platform" oe:value="ns1000v"/>
 <Property oe:key="com.citrix.netscaler.orch_env" oe:value="cisco-orch-env"/>
 <Property oe:key="com.citrix.netscaler.mgmt.ip" oe:value="10.102.38.82"/>
 <Property oe:key="com.citrix.netscaler.mgmt.netmask"  oe:value="255.255.255.0"/>
 <Property oe:key="com.citrix.netscaler.mgmt.gateway"  oe:value="10.102.38.1"/>
</PropertySection>
</Environment>
```

Provisioning the NetScaler Virtual Appliance by using OpenStack Dashboard

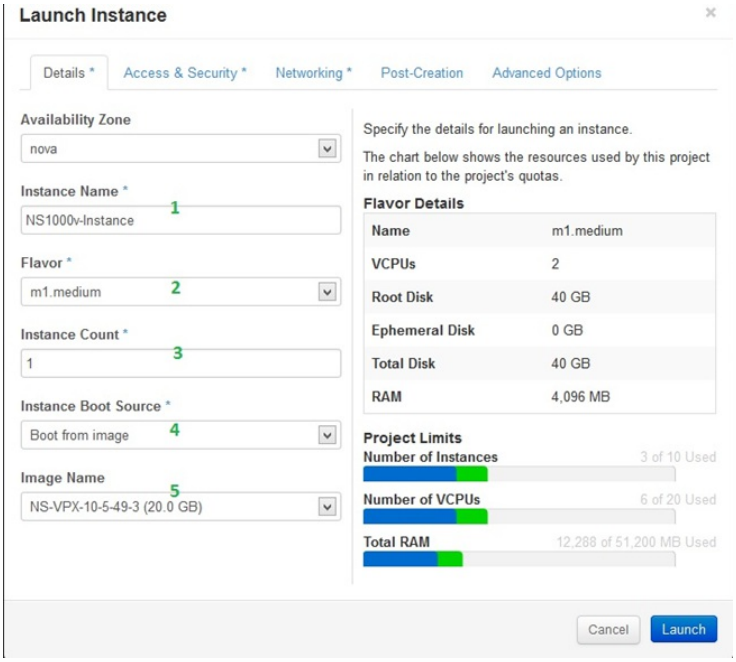
Updated: 2015-02-26

You can provisioning NetScaler in an OpenStack environment using the OpenStack dashboard.

1. Log in to the OpenStack dashboard.
2. In the Project panel on the left hand side of the dashboard, select Instances.
3. In the Instances panel, click Launch Instance to open the Instance Launching Wizard.



4. In the Launch Instance wizard, fill in the details, like:
    1. Instance Name
    2. Instance Flavor
    3. Instance Count
    4. Instance Boot Source
    5. Image Name



5. Click on the Post Creation tab in the wizard. In the Customization Script, add the content of the userdata file. The userdata file contains the IP address, Netmask and Gateway details of the NetScaler instance.
6. Click Launch.

Provisioning the NetScaler Virtual Appliance by using the Virtual Machine Manager

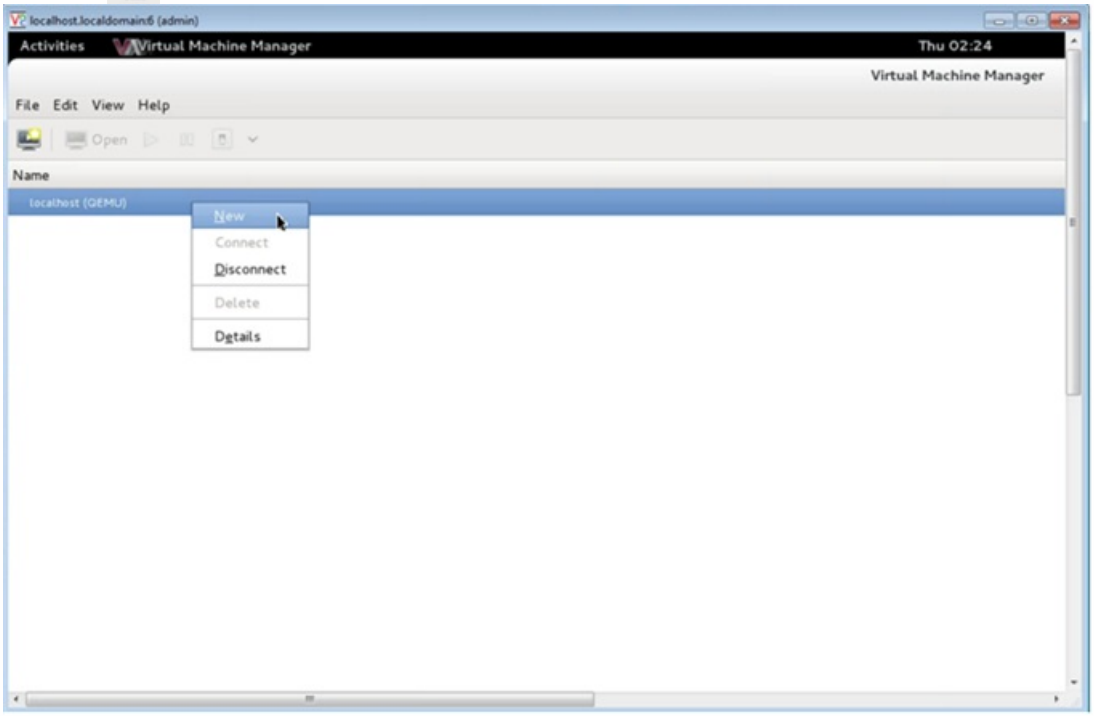
Mar 10, 2015

The Virtual Machine Manager is a desktop tool for managing VM Guests. It enables you to create new VM Guests and various types of storage, and manage virtual networks. You can access the graphical console of VM Guests with the built-in VNC viewer and view performance statistics, either locally or remotely.

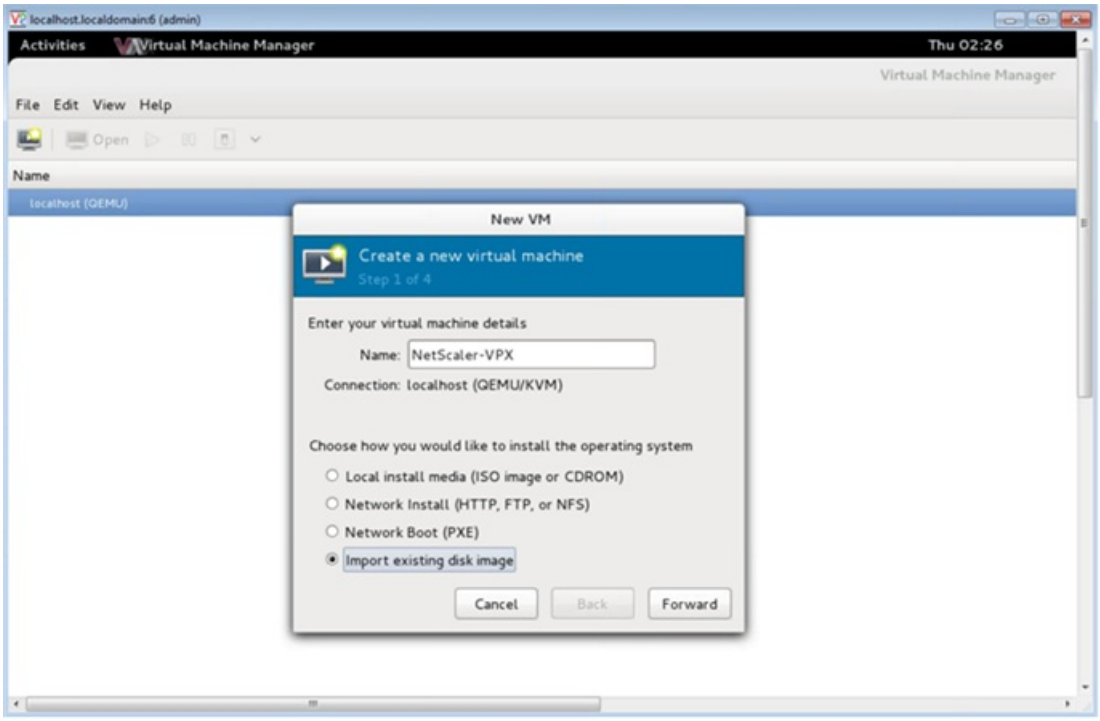
After installing your preferred Linux distribution, with KVM virtualization enabled, you can proceed with provisioning virtual machines.

To provision a NetScaler VPX by using Virtual Machine Manager

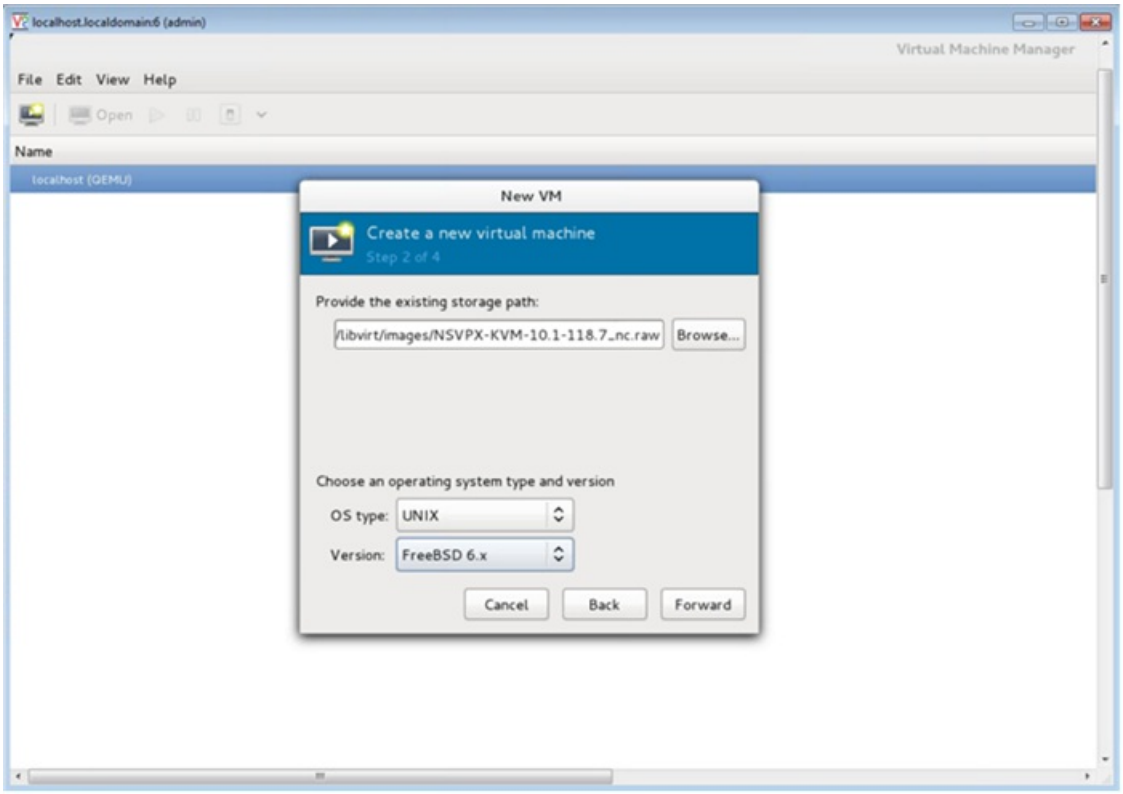
1. Open the Virtual Machine Manager (**Application** > **System Tools** > **Virtual Machine Manager**) and enter the logon credentials in the **Authenticate** window.
2. Click the  icon or right-click **localhost (QEMU)** to create a new NetScaler VPX instance.



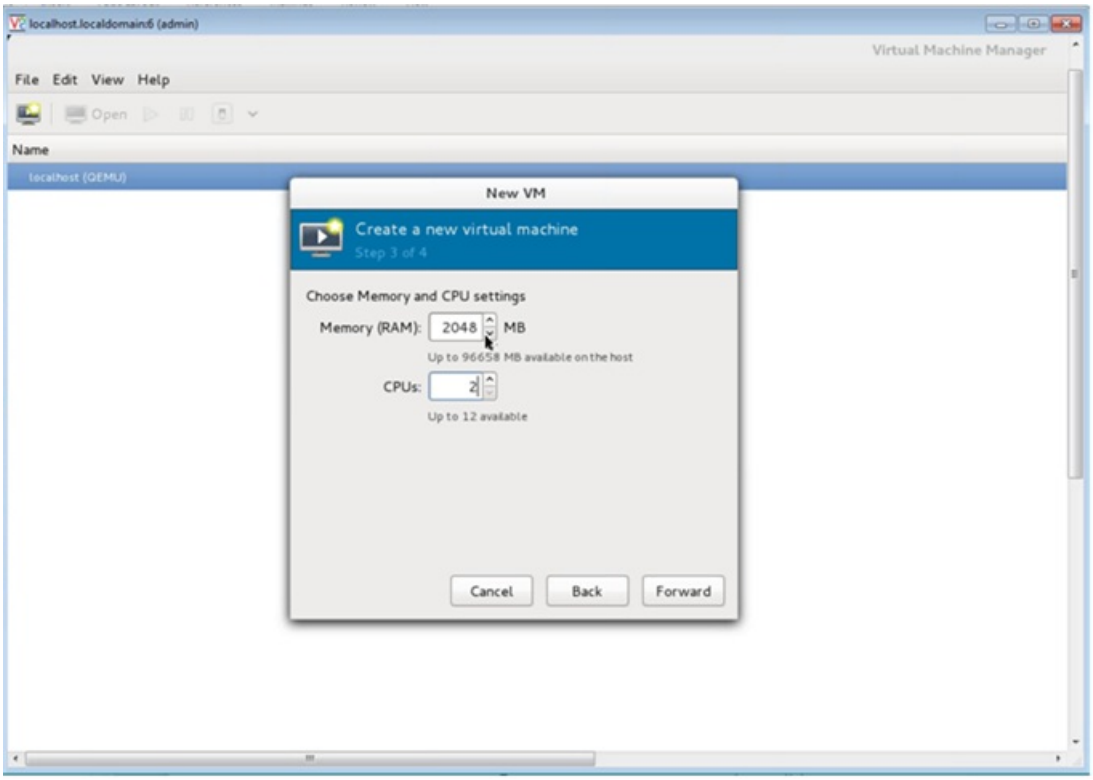
3. In the **Name** text box, enter a name for the new VM (for example, NetScaler-VPX).
4. In the **New VM** window, under "Choose how you would like to install the operating system," select **Import existing disk image**, and then and click **Forward**.



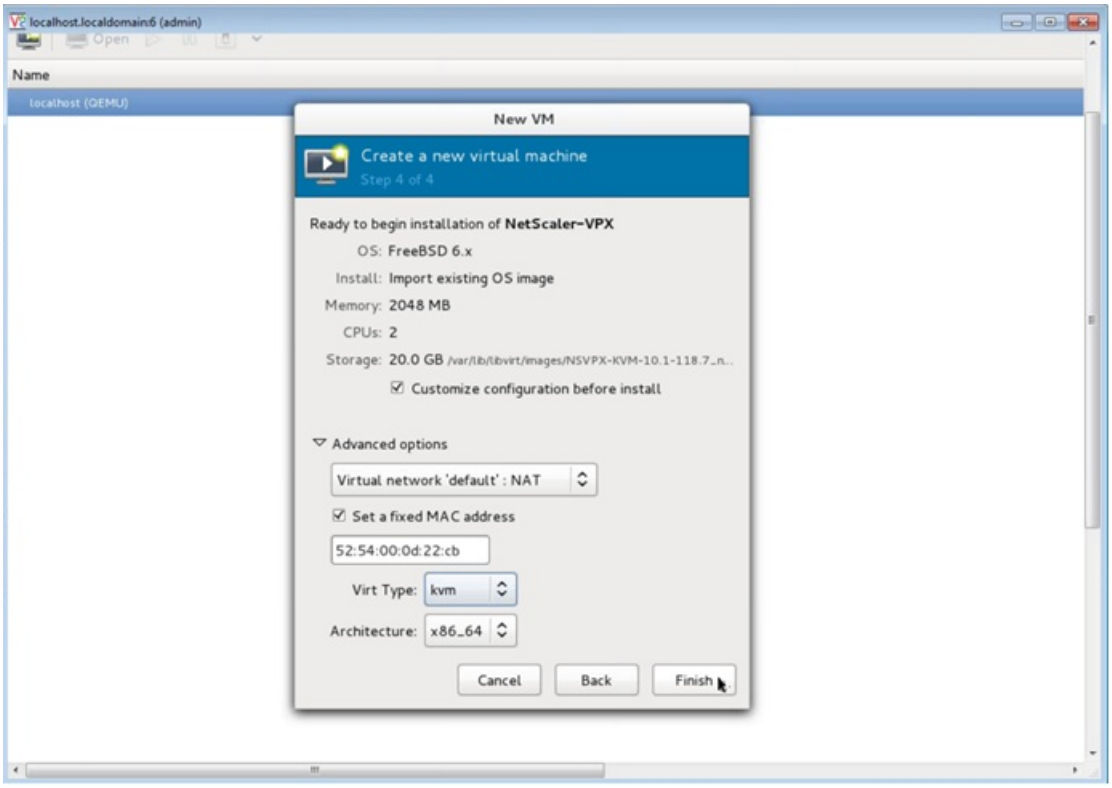
5. In the **Provide the existing storage path** field, navigate the path to the image. Choose the OS type as UNIX and Version as FreeBSD 6.x. Then, click **Forward**.



6. Under "Choose Memory and CPU settings," select the following settings, and then click **Forward**:
   - Memory (RAM)— 2048 MB
   - CPUs— 2

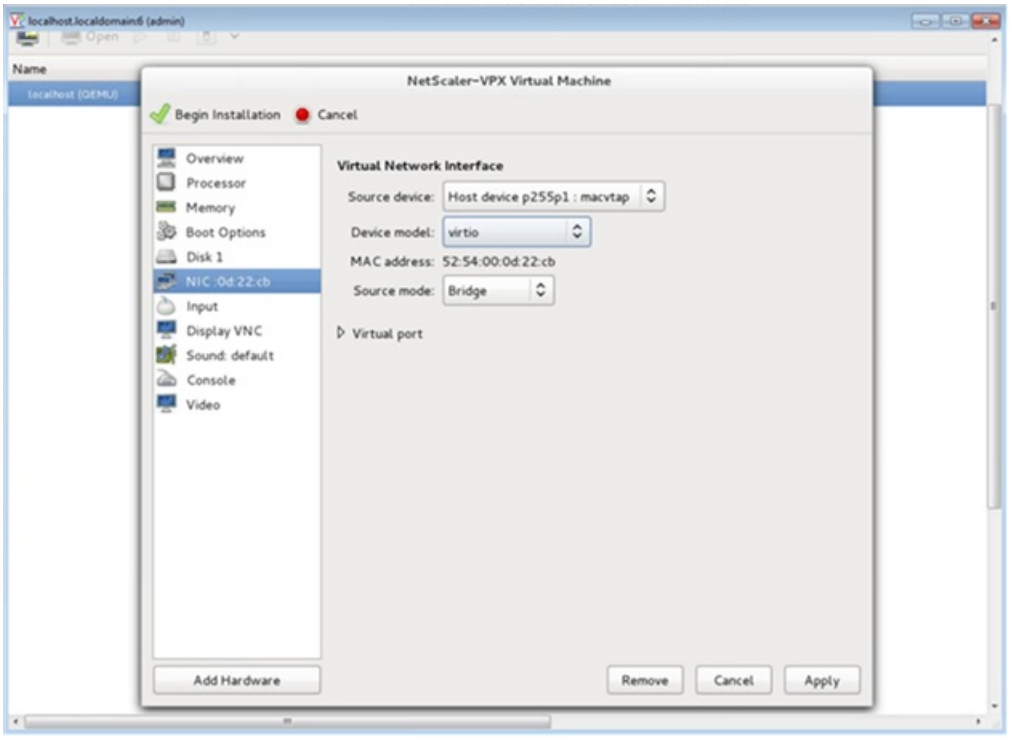


7. Select the **Customize configuration before install** check box. Optionally, under "Advanced options," you can you can customize the MAC address. Make sure the **Virt Type** selected is kvm and the Architecture selected is x86_64. Click **Finish**.



8. Select a NIC and provide the following configuration:
   - Source device— ethX macvtap or Bridge
   - Device model— virtio

- Source mode— Bridge



9. Click **Apply**, and then click **Begin Installation**. After you have provisioned the NetScaler VPX on KVM, you can add additional interfaces

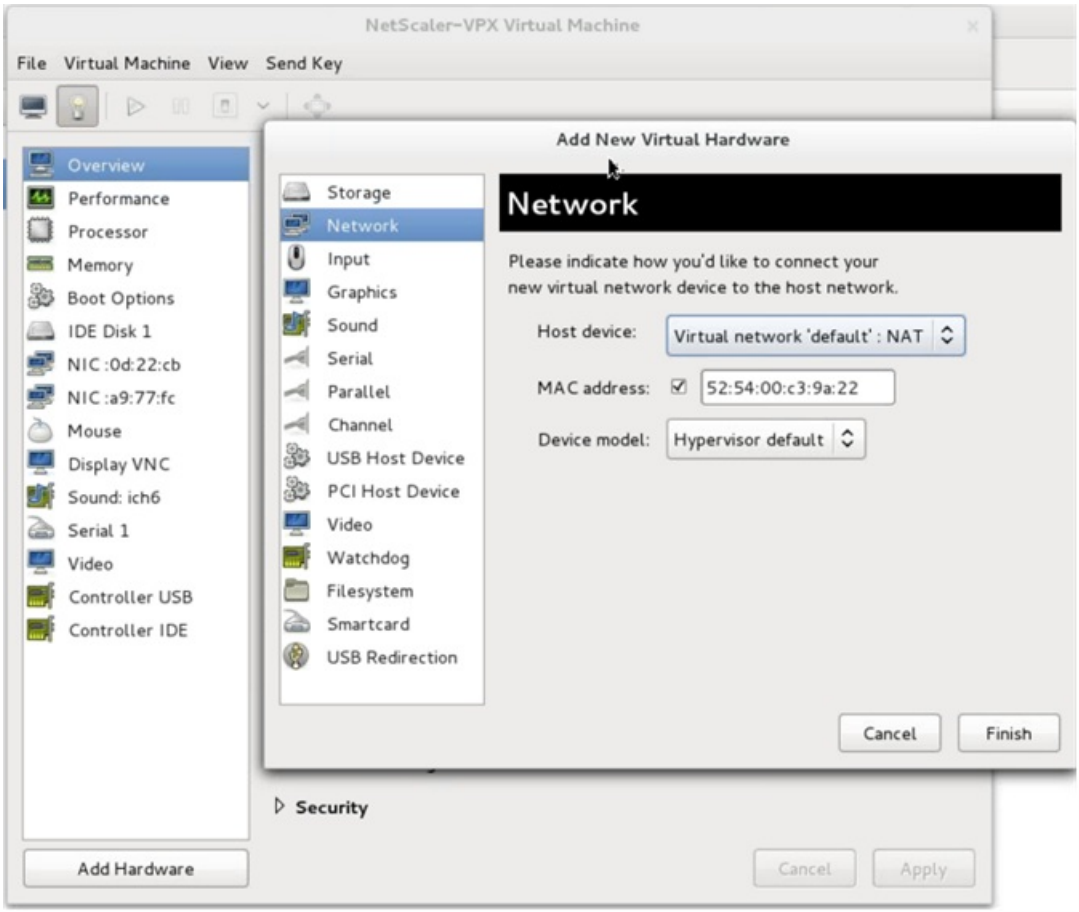
Adding Additional Interfaces to NetScaler VPX by using Virtual Machine Manager

Updated: 2015-03-11

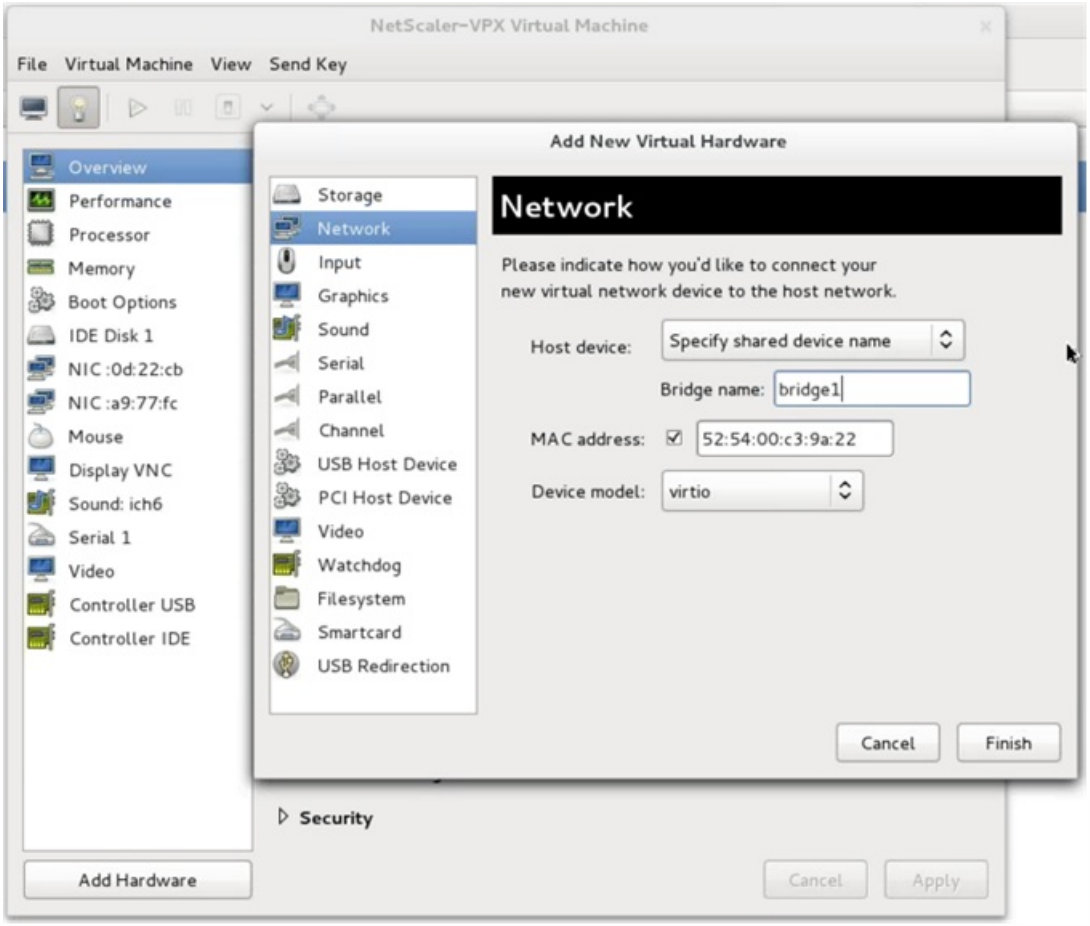
After you have provisioned the NetScaler VPX on KVM, you can add additional interfaces.

To add additional interfaces

1. Shut down the NetScaler VPX instance running on the KVM.
2. Right-click the VPX instance and choose **Open** from the pop-up menu.
3. Click the  icon in the header to view the virtual hardware details.
4. Click **Add Hardware**. In the **Add New Virtual Hardware window**, select **Network** from the navigation menu.

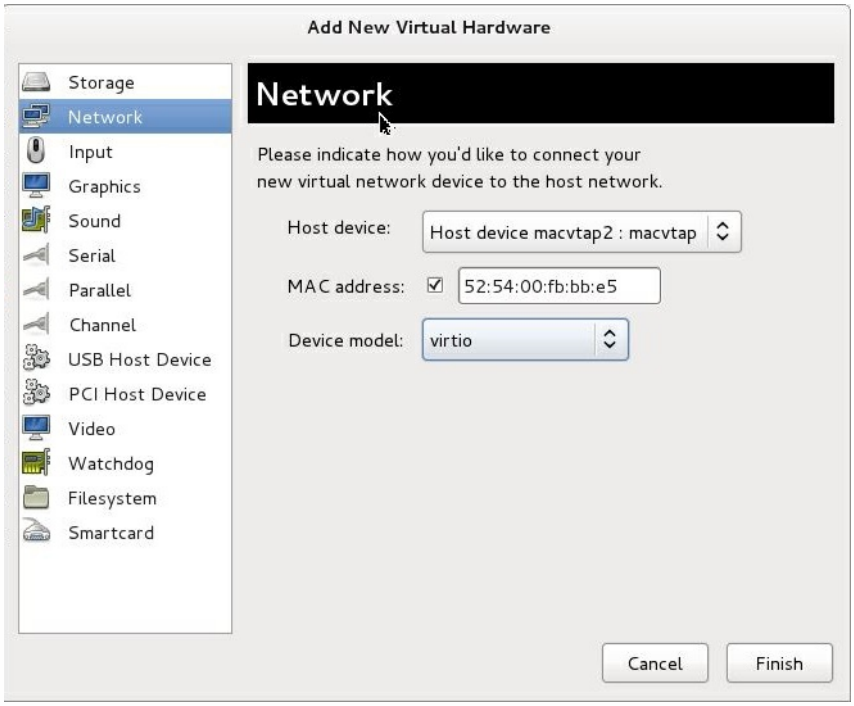


5. In **Host Device** field, select the physical interface type. The host device type can be either Bridge or MacVTap. In case of MacVTap, four modes possible are VEPA, Bridge, Private and Pass-through.
    1. For Bridge
        1. Host device— Select the "Specify shared device name" option.
        2. Provide the Bridge name that is configured in the KVM host.
           Note: Make sure that you have configured a Linux bridge in the KVM host, bound the physical interface to the bridge, and put the bridge in the UP state.

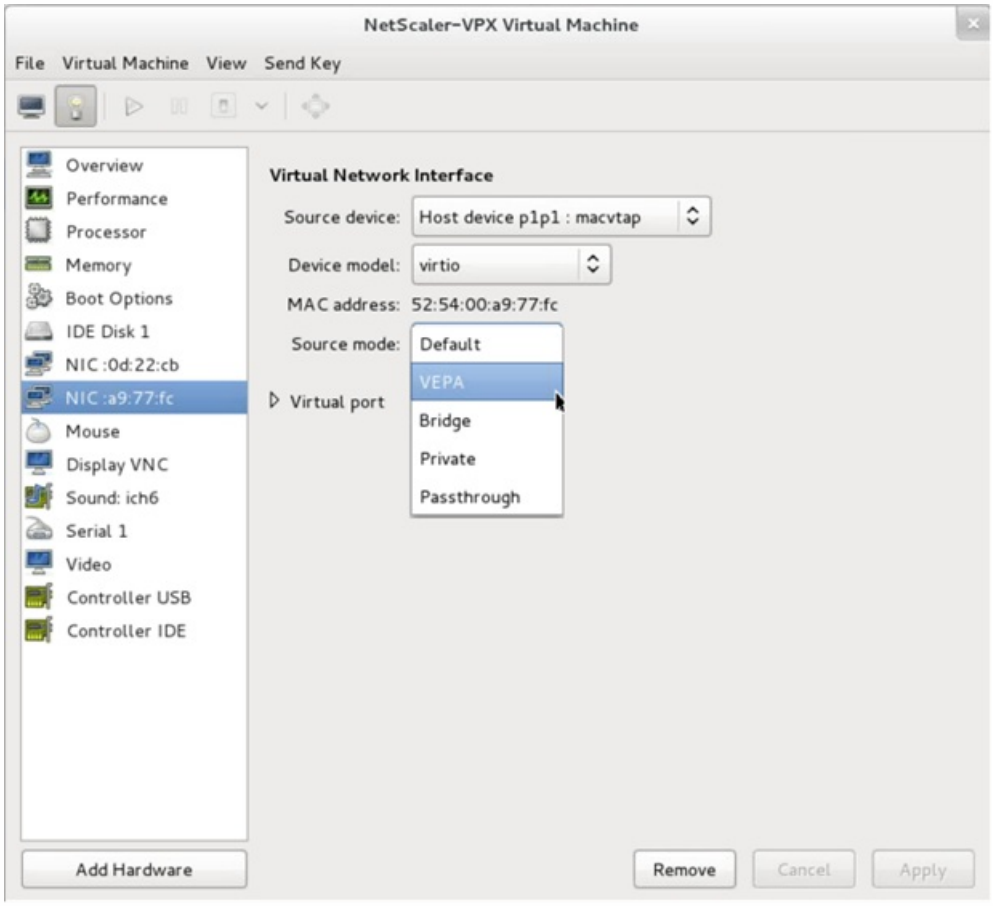


3. Device model—virtio.
4. Click Finish.

2. For MacVTap
   1. Host device—Select the physical interface from the menu.
   2. Device model—virtio.



3. Click Finish. You can view the newly added NIC in the navigation pane.



4. Select the newly added NIC and select the Source mode for this NIC. The available modes are VEPA, Bridge, Private, and Passthrough. For more details on the interface and modes, see Source Interface and Modes.
5. Click Apply.
6. Start the NetScaler VPX VM.

Provisioning the NetScaler Virtual Appliance by using the virsh Program

Mar 10, 2015

The virsh program is a command line tool for managing VM Guests. Its functionality is similar to that of Virtual Machine Manager. It enables you to change a VM Guest's status (start, stop, pause, and so on), to set up new Guests and devices, and to edit existing configurations. The virsh program is also useful for scripting VM Guest management operations.

To provision NetScaler VPX by using the virsh program

1. Use the tar command to untar the the NetScaler VPX package. The NSVPX-KVM-*_nc.tgz package contains following components:
   - The Domain XML file specifying VPX attributes [NSVPX-KVM-*_nc.xml]
   - Check sum of NS-VM Disk Image [Checksum.txt]
   - NS-VM Disk Image [NSVPX-KVM-*_nc.raw]

   Example:
   ```
   tar -xvzf NSVPX-KVM-10.1-117_nc.tgz
   NSVPX-KVM-10.1-117_nc.xml
   NSVPX-KVM-10.1-117_nc.raw
   checksum.txt
   ```
2. Copy the NSVPX-KVM-*_nc.xml XML file to a file named <DomainName>-NSVPX-KVM-*_nc.xml. The <DomainName> is also the name of the virtual machine. Example:
   ```
   cp NSVPX-KVM-10.1-117_nc.xml NetScaler-VPX-NSVPX-KVM-10.1-117_nc.xml
   ```
3. Edit the <DomainName>-NSVPX-KVM-*_nc.xml file to specify the following parameters:
   - name— Specify the name.
   - mac— Specify the MAC address.
     Note: The domain name and the MAC address have to be unique.
   - sourcefile— Specify the absolute disk-image source path. The file path has to be absolute. In this example, the disk image is at the following location: /root/NSVPX-KVM-10.1-117_nc.raw.

   Example:
   ```
   <name>NetScaler-VPX</name>
         <mac address='52:54:00:29:74:b3'/>
         <source file='/root/NSVPX-KVM-10.1-117_nc.raw'/>
   ```
4. Edit the <DomainName>-NSVPX-KVM-*_nc.xml file to configure the networking details:
   - source dev— specify the interface.
   - mode— specify the mode. The default interface is **Macvtap Bridge**.

   Example: Mode: MacVTap Bridge Set target interface as ethx and mode as bridge Model type as virtio
   ```
    <interface type='direct'>
      <mac address='52:54:00:29:74:b3'/>
      <source dev='eth0' mode='bridge'/>
      <target dev='macvtap0'/>
      <model type='virtio'/>
      <alias name='net0'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
    </interface>
   ```

Here, eth0 is the physical interface attached to the VM.

5. Define the VM attributes in the <DomainName>-NSVPX-KVM-*_nc.xml file by using the following command: virsh define <DomainName>-NSVPX-KVM-*_nc.xml Example:
   virsh define NS-VPX-NSVPX-KVM-10.1-117_nc.xml
6. Start the VM by entering following command: virsh start [<DomainName> | <DomainUUID>] Example:
   virsh start NetScaler-VPX
7. Connect the Guest VM through the console virsh console [<DomainName> | <DomainUUID> |<DomainID>] Example:
   virsh console NetScaler-VPX

Adding Additional Interfaces to NetScaler VPX using virsh Program

Updated: 2015-03-09

After you have provisioned the NetScaler VPX on KVM, you can add additional interfaces.

To add additional interfaces

1. Shut down the NetScaler VPX instance running on the KVM.
2. Edit the <DomainName>-NSVPX-KVM-*_nc.xml file using the command: virsh edit [<DomainName> | <DomainUUID>]
3. In the <DomainName>-NSVPX-KVM-*_nc.xml file, append the following parameters:
   1. **For MacVTap**
      - Interface type— Specify the interface type as 'direct'.
      - Mac address— Specify the Mac address and make sure the MAC address is unique across the interfaces.
      - source dev— Specify the interface name.
      - mode— Specify the mode; the modes supported are - Bridge, VEPA, Private, and Pass-through
      - model type— Specify the model type as virtio

      Example:

      Mode: MacVTap Pass-through

      Set target interface as ethx, Mode as bridge, and model type as virtio

      ```
      <interface type='direct'>
          <mac address='52:54:00:29:74:b3'/>
          <source dev='eth1' mode='passthrough'/>
          <model type='virtio'/>
       </interface>
      ```

      Here eth1 is the physical interface attached to the VM.

   2. **For Bridge Mode**

      Note: Make sure that you have configured a Linux bridge in the KVM host, bound the physical interface to the bridge, and put the bridge in the UP state.
      - Interface type— Specify the interface type as 'bridge'.
      - Mac address— Specify the Mac address and make sure the MAC address is unique across the interfaces.
      - source bridge— Specify the bridge name.
      - model type— Specify the model type as virtio

      Example: Bridge Mode

      ```
      <interface type='bridge'>
      ```



```
    <mac address='52:54:00:2d:43:a4'/>
    <source bridge='br0'/>
    <model type='virtio'/>
</interface>
```

Managing the NetScaler Guest VMs

Sep 03, 2013

You can use the Virtual Machine Manager and the virsh program to perform management tasks such as starting or stopping a VM Guest, setting up new guests and devices, editing existing configurations, and connecting to the graphical console through Virtual Network Computing (VNC).

Updated: 2013-09-04

Listing the VM Guests

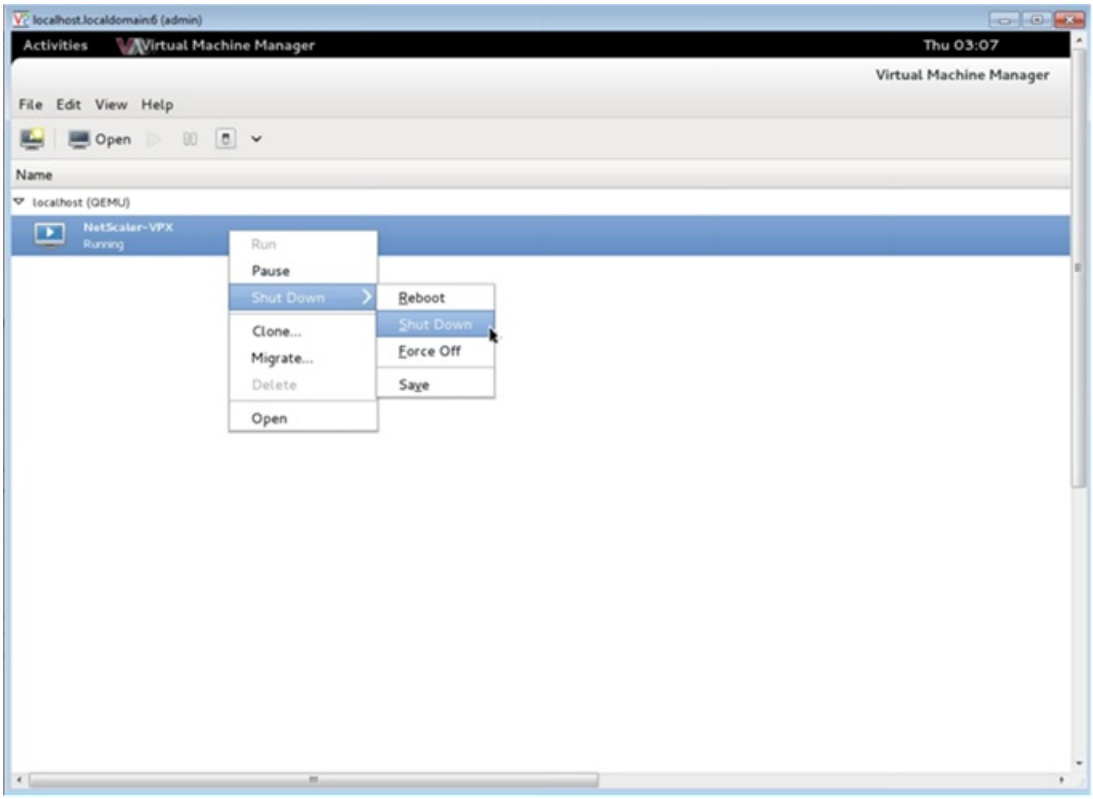
The main Window of the Virtual Machine Manager displays a list of all the VM Guests for each VM host server it is connected to. Each VM Guest entry contains the virtual machine's name, along with its status (Running, Paused, or Shutoff) displayed as icon.

Opening a Graphical Console

Opening a Graphical Console to a VM Guest enables you to interact with the machine like you would with a physical host through a VNC connection. To open the graphical console in the Virtual Machine Manager, right-click the VM Guest entry and select the Open option from the pop-up menu.

Starting and Shutting Down a Guest

You can start or stop a VM Guest from the Virtual Machine Manager. To change the state of the VM, right-click the VM Guest entry and select Run or one of the Shut Down options from the pop-up menu.



Rebooting a Guest

You can reboot a VM Guest from the Virtual Machine Manager. To reboot the VM, right-click the VM Guest entry, and then

select Shut Down > Reboot from the pop-up menu.

Deleting a Guest

Deleting a VM Guest removes its XML configuration by default. You can also delete a guest's storage files. Doing so completely erases the guest.

1. In the Virtual Machine Manager, right-click the VM Guest entry.
2. Select Delete from the pop-up menu. A confirmation window opens.
   Note: The Delete option is enabled only when the VM Guest is shut down.
3. Click Delete.
4. To completely erase the guest, delete the associated .raw file by selecting the Delete Associated Storage Files check box.

Updated: 2013-09-04

Listing the VM Guests and their current states

To use virsh to display information about the Guests

```
virsh list --all
```

The command output displays all domains with their states.

Example Output:

```
Id Name                 State
----------------------------------
 0 Domain-0             running
 1 Domain-1             paused
 2 Domain-2             inactive
 3 Domain-3             crashed
```

Opening a virsh Console

Connect the Guest VM through the console

```
virsh console [<DomainID> | <DomainName> | <DomainUUID>]
```

Example:

```
virsh console NetScaler-VPX
```

Starting and Shutting Down a Guest

Guests can be started using the DomainName or Domain-UUID.

```
virsh start [<DomainName> | <DomainUUID>]
```

Example

```
virsh start NetScaler-VPX
```

To shut down a guest:

```
virsh shutdown [<DomainID> | <DomainName> | <DomainUUID>]
```

Example:

```
virsh shutdown NetScaler-VPX
```

Rebooting a Guest

```
virsh reboot [<DomainID> | <DomainName> | <DomainUUID>]
```

Example:

virsh reboot NetScaler-VPX

Deleting a Guest

To delete a Guest VM you need to shut-down the Guest and un-define the <DomainName>-NSVPX-KVM-*_nc.xml before you run the delete command.

virsh shutdown [<DomainID> | <DomainName> | <DomainUUID>]
virsh undefine [<DomainName> | <DomainUUID>]

Example:

virsh shutdown NetScaler-VPX
virsh undefine NetScaler-VPX

Note: The delete command doesn't remove disk image file which needs to be removed manually.

Installing NetScaler VPX on AWS

Jan 29, 2014

You can now launch an instance of Citrix® NetScaler VPX within Amazon Web Services (AWS). NetScaler VPX is available as an Amazon Machine Image (AMI) in AWS marketplace. NetScaler VPX on AWS enables customers to leverage AWS Cloud computing capabilities and use NetScaler load balancing and traffic management features for their business needs. NetScaler on AWS supports all the traffic management features of a physical NetScaler appliance. NetScaler instances running in AWS can be deployed as standalone instances or in HA pairs.

Updated: 2014-05-13

AWS offers different types of web services, such as Amazon Simple Storage Services (S3), Amazon Elastic Cloud Compute (EC2), and Amazon Virtual Private Cloud (VPC). Amazon VPC allows you to run AWS resources (for example, EC2 instances) in a private, virtual network. Amazon EC2 instances are available as instance types that map to hardware archetypes on the basis of factors such as number of EC2 Compute Units (ECU), number of virtual cores, and memory size.

The NetScaler VPX AMI is packaged as an EC2 instance that is launched within an AWS VPC. The VPX AMI instance requires a minimum of 2 virtual CPUs and 2 GB of memory. An EC2 instance launched within an AWS VPC can also provide the multiple interfaces, multiple IP addresses per interface, and public and private IP addresses needed for VPX configuration. Currently, on Amazon AWS, VPX can be launched only within a VPC, because each VPX instance requires at least three IP addresses. (Although VPX on AWS can be implemented with one or two elastic network interfaces, Citrix recommends three network interfaces for a standard VPX on AWS installation.) AWS currently makes multi-IP functionality available only to instances running within an AWS VPC. A VPX instance in a VPC can be used to load balance servers running in EC2 instances.

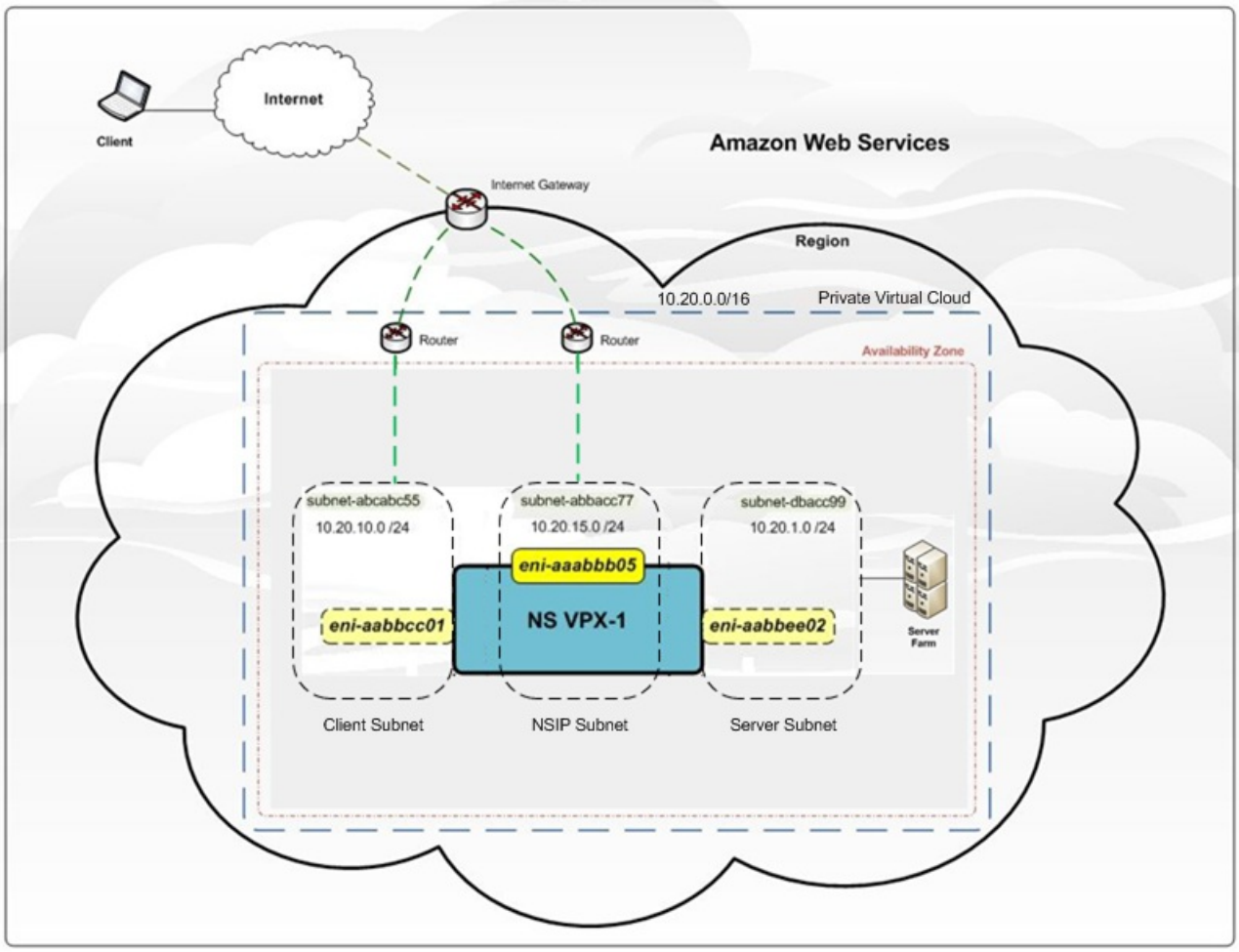
An Amazon VPC allows you to create and control a virtual networking environment, including your own IP address range, subnets, route tables, and network gateways.

Note: By default, you can create up to 5 VPC instances per AWS region for each AWS account. You can request higher VPC limits by submitting Amazon's request form (http://aws.amazon.com/contact-us/vpc-request/).

VPX on AWS Architecture

An EC2 instance of NetScaler VPX (AMI image) is launched within the AWS VPC. The following figure shows a typical VPX on AWS deployment.

Figure 1. VPX on AWS Architecture



The figure shows a simple topology of an AWS VPC with a NetScalerVPX deployment. The AWS VPC has:

1. A single Internet gateway to route traffic in and out of the VPC.
2. Network connectivity between the Internet gateway and the Internet.
3. Three subnets, one each for management, client, and server.
4. Network connectivity between the Internet gateway and the two subnets (management and client).
5. A single NetScaler VPX deployed within the VPC. The VPX instance has three Elastic Network Interfaces (ENIs), one attached to each subnet.

Supported EC2 instances

The NetScaler AMI can be launched on any of the following EC2 instance types:

- m3.large
- m3.xlarge
- m3.2xlarge

For more information about Amazon EC2 instances, see:
http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/instance-types.html

Updated: 2014-05-13

The following table lists the EC2 instance types and corresponding number of supported ENIs and number of private IP addresses per ENI.

Table 1. EC2 Support for ENIs and IP Addresses

Instance Name	Number of ENIs	Private IP Addresses per ENI
m3.large	3	10
m3.xlarge	4	15
m3.2xlarge	4	30

Limitations and Usage Guidelines

May 13, 2014

- The clustering feature is not supported for VPX.
- For HA to work as expected, associate a dedicated NATing device to management Interface or associate EIP to NSIP. For more information on NAT, in the AWS documentation, see NAT Instances.
- Data traffic and management traffic should be segregated by using ENIs belonging to different subnets.
- Only the NSIP address should be present on the management ENI.
- To send traffic to VIPs, you must enable MAC Based Forwarding (MBF) and Use Subnet IP (USNIP) modes on VPX.
- If a NAT instance is used for security instead of assigning an EIP to the NSIP, appropriate VPC level routing changes are required. For instructions on making VPC level routing changes, in the AWS documentation, see "Scenario 2: VPC with Public and Private Subnets."
- A VPX instance can be moved from one EC2 instance type to another (for example, from m3.large to an m3.xlarge).
- For storage options for VPX on AWS, Citrix recommends EBS, because it is durable and the data is available even after it is detached from instance.
- Dynamic addition of ENIs to VPX is not supported. You have to restart the VPX instance to apply the update. Citrix recommends you to stop the standalone or HA instance, attach the new ENI, and then restart the instance.
- You can assign multiple IP addresses to an ENI. The maximum number of IP addresses per ENI is determined by the EC2 instance type, see EC2 Support for ENIs and IP Addresses.
- Citrix recommends that you avoid using the enable and disable interface commands on NetScaler VPX interfaces.
- Due to Amazon AWS limitations, these features are not supported:
  - IPV6
  - Gratuitous ARP(GARP)
  - L2 mode
  - Tagged VLAN
  - Dynamic Routing
  - Virtual MAC (VMAC)

Launching the NetScaler VPX for AWS AMI

May 14, 2014

You can launch a Citrix NetScaler VPX AMI within an Amazon Web Services (AWS) Virtual Private Cloud (VPC) in one of two ways:

1. Using the Amazon GUI and CLI toolkit.
2. Using a Citrix authored CloudFormation template.
3. Using the Amazon 1-Click launch.

Note: The following are the default administrator credentials to access a NetScaler VPX instance:

- Username—nsroot
- Password—The default password for the nsroot account is set to the AWS instance-ID of the NetScaler VPX instance. For a high availability configuration between two NetScaler VPX instances, the nsroot password of the secondary node is set to that of the primary node after the HA configuration synchronization.

Updated: 2014-05-13

To launch a NetScaler VPX AMI within an Amazon Web Services (AWS) Virtual Private Cloud (VPC) by using the Amazon GUI and CLI toolkit, you need:

- An AWS account
- An AWS Virtual Private cloud (VPC)
- The AWS API toolkit (if creating a VPX instance with three or more ENIs).
- An IAM account

Creating an AWS Account

To launch a NetScaler VPX AMI in an Amazon Web Services (AWS) Virtual Private Cloud (VPC), you need an AWS account. You can create an AWS account for free at www.aws.amazon.com.

Creating an AWS Virtual Private Cloud (VPC)

Citrix recommends at least three IP addresses for a NetScaler instance. Currently, the only support that AWS provides for instances with multiple IP addresses is for instances within an AWS VPC.

To create an AWS VPC, first launch the AWS GUI console. For instructions for using the AWS GUI console, see http://docs.amazonwebservices.com/AmazonVPC/latest/GettingStartedGuide/GetStarted.html?r=2900.

To create an AWS VPC

1. Use the VPC with a Single Public Subnet Only option to create a new AWS VPC in an AWS availability zone.
2. Create additional subnets within the AWS VPC. Citrix recommends that you create at least three subnets, of the following types:
   - One subnet for NetScaler management traffic. You place the NetScaler management IP(NSIP) on this subnet.
   - One or more subnets for client-access (user-to-NetScaler) traffic, through which clients connect to one or more virtual IP (VIP) addresses assigned to NetScaler load balancing virtual servers.
   - One or more subnets for the server-access (NetScaler-to-server) traffic, through which your servers connect to NetScaler-owned subnet IP (SNIP) addresses.
     For more information about NetScaler load balancing and virtual servers, virtual IP addresses (VIPs), and subnet IP addresses (SNIPs), see: Getting Started with Citrix NetScaler.

   Note:
   - All subnets should be in the same availability zone.
   - You can launch a NetScaler AMI in an AWS VPC with a single subnet. In this configuration, the management traffic, client-side traffic, and server-side traffic all use the same subnet, and high availability (HA) cannot be configured.
   - You can launch the NetScaler AMI into an AWS VPC with two subnets. In this configuration, one subnet is used for management traffic, and the other subnet is used for both client-side and server-side traffic. This topology supports NetScaler HA.
3. Create an Internet gateway and attach it to the VPC instance.
4. Create routing tables for all traffic flowing into or out of the VPC. You need routes for access to the NSIP and to any client-facing VIP addresses. Traffic leaving the VPC must be routed through the Internet Gateway of the AWS VPC.
   Note:
   - Make sure that you associate management and client subnets with the routing table.
   - Add a default route to the routing table for the traffic flowing out of the VPC. Set the Destination to 0.0.0.0/0, and the Target as the Internet gateway address.
5. Create a security group and open the required ports.

Setting-up the AWS API Toolkit

The AWS GUI console does not allow you to launch instances with more than two ENIs. For a standard deployment, you have to create at least three ENIs for a VPC instance (though it is possible to launch a NetScaler AMI with one or two ENIs). To create three or more ENIs for a NetScaler instance, you must use the AWS CLI. To use the AWS CLI, you must install the AWS API toolkit.

The AWS API toolkit is available for download at http://aws.amazon.com/developertools/351/. To install the AWS API toolkit, complete the following tasks on a Windows or Linux machine:

1. Download the AWS API Toolkit.
2. Download X.509 certificate files and X.509 private key file.
3. Download the private key.
4. Convert the downloaded private key (.pem file) for SSH connectivity.
5. Configure the AWS API Toolkit environment on your Windows or Linux computer.

To download the AWS API toolkit

1. In a web browser, open the following website: http://aws.amazon.com/developertools/351/.

2. On the Amazon EC2 API Tools page, in the Download section, click Download the Amazon EC2 API Tools.
3. Save the file, ec2-api-tools.zip, to a local disk and use a file compression utility (for example, WinZip) to extract the files.

To download the X.509 certificate file and X.509 private key file

1. In your browser, open the following website: http://aws.amazon.com/.
2. Click My Account/Console, and then click Security Credentials.
3. On the Amazon Web Services Sign in page, use your Amazon account credentials to sign in.
4. On the Security Credentials page, in the Access Credentials section, on the X.509 Certificates tab, click Create a New Certificate.
5. In the X509 Certificate Created dialog box, Click Download Private Key File and save the private key file to a secure folder on your local drive.
6. Click Download X.509 Certificate and save the certificate to a secure folder on your local drive.
7. Click Close.

Note: The Private Key File can be downloaded only at the time of creating a certificate. However, you can download the certificate at any time after creating it.

To download private key for SSH connectivity

1. In your browser, open the following website: http://aws.amazon.com/ .
2. Click **My Account/Console.**
3. On the **Amazon Web Services Sign in** page, use your Amazon account credentials to sign in.
4. In the **Service** pane, in **Amazon Web Services**, click **EC2**.
5. In the **Navigation** section, in **Network and Security**, click **Key Pairs**.
6. In the **Key Pairs** pane, click **Create Key Pair**.
7. In the **Create Key Pair** dialog box, type the name for key pair and click **Create**.
8. Download the Key Pair to the local disk and click **Close**.

To convert the downloaded private key for SSH connectivity

For SSH connections from a management machine using Putty, you must convert the .pem file (Private Key) into .ppk file. The .ppk file is the private key for SSH connections to the NetScaler VPX instance hosted in the AWS environment. To convert the .pem file to a .ppk file, use the Putty application's PuttyGen utility. Make sure that the key pairs and certificate files are stored in an unshared and secured directory. After the conversion, you can use SSH to securely connect to the management address of the VPX on AWS instance.

To configure the AWS API Toolkit environment on a Windows machine

1. Move the certificate files to an unshared folder (for example, aws-ec2-api-tools).
2. Move the extracted AWS API toolkit folder to the unshared folder (for example, the aws-ec2-api-tools folder created in example in Step 1).
3. Create a batch file to configure the specific AWS environment in the unshared folder (aws-ec2-api-tools if you used the example in the preceding two steps). Following is an example of the batch file. The file location used in this example is C:\aws-vpc-config\ and the file name is set-aws-environment.bat.

```
rem Setup Amazon EC2 Command-Line Tools

set JAVA_HOME="C:\Program Files\Java\jre7\"

set EC2_HOME="C:\aws-ec2-api-tools\"

set PATH=%PATH%;%EC2_HOME%\bin
```



```
set EC2_PRIVATE_KEY=C:\aws-ec2-security-files\pk-3T6ACCLBEDGD3O3SMAM7YDI76VP5HXSU.pem

set EC2_CERT=C:\aws-ec2-security-files\cert-3T6ACCLBEDGD3O3SMAM7YDI76VP5HXSU.pem

set EC2_URL=https://<aws-region>.ec2.amazonaws.com
```

4. Open the command prompt and run the batch file. For the file in the above example, type:
   C:\aws-vpc-config> set-aws-environment.bat
5. Run the ec2ver command to verify that the AWS toolkit is installed properly. For example:
   C:\aws-vpc-config>ec2ver 1.5.6.1 2012-06-15

To configure the AWS API Toolkit on a Linux machine

1. Move the certificate files to an unshared folder (for example, aws-ec2-api-tools).
2. Move the extracted AWS API toolkit folder to the unshared folder (for example, the aws-ec2-api-tools folder created in example in Step 1).
3. Create a shell script to configure the specific AWS environment in the unshared folder (aws-ec2-api-tools if you used the example in the preceding two steps). Following is an example of the batch file. In this example, the file location used is C:\aws-vpc-config\ and the file name used is set-aws-environment.bat.

```
# Setup Amazon EC2 Command-Line Tools

export EC2_HOME=~/ec2-api-tools-1.5.6.0

export EC2_URL= https://us-east-1.ec2.amazonaws.com

export PATH=$EC2_HOME/bin:/usr/bin:$/usr/sbin:/usr/local/sbin:/sbin

export EC2_PRIVATE_KEY=~/pk-XOX3NS2UPZL6BGLFO7PM5OGLYBDPBUCB.pem

export EC2_CERT=~/cert-XOX3NS2UPZL6BGLFO7PM5OGLYBDPBUCB.pem

export JAVA_HOME=/usr

export PS1="AWS PROMPT >"
```

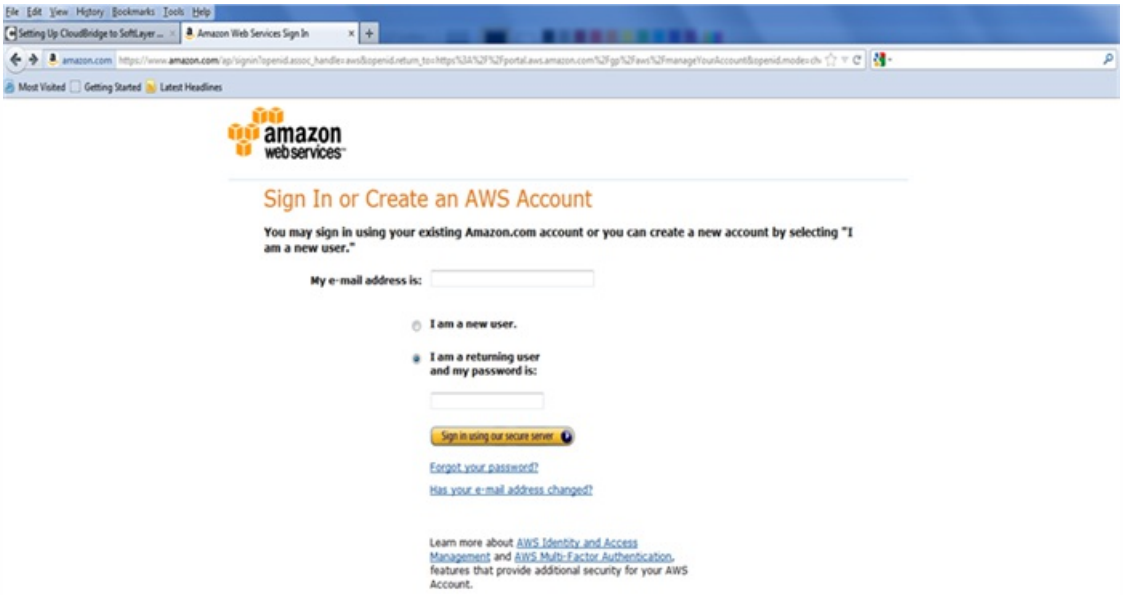
4. Run the ec2ver command to verify that the AWS toolkit is installed properly. For example:

```
AWS PROMPT >ec2ver

1.5.6.1 2012-06-15
```

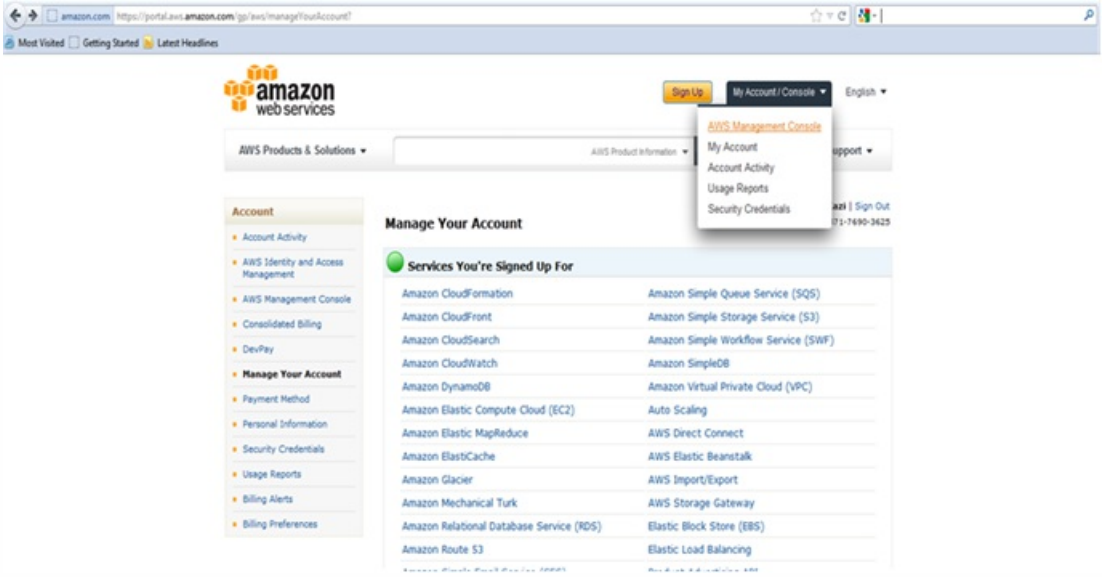
Creating an IAM Account

Before you launch the VPX AMI instance, you have to create a new IAM user account with the Access and Secret keys. The Access and Secret key credentials from the new IAM user are required for launching the NetScaler AMI instance. To create a new IAM user for NetScaler, complete the following steps.

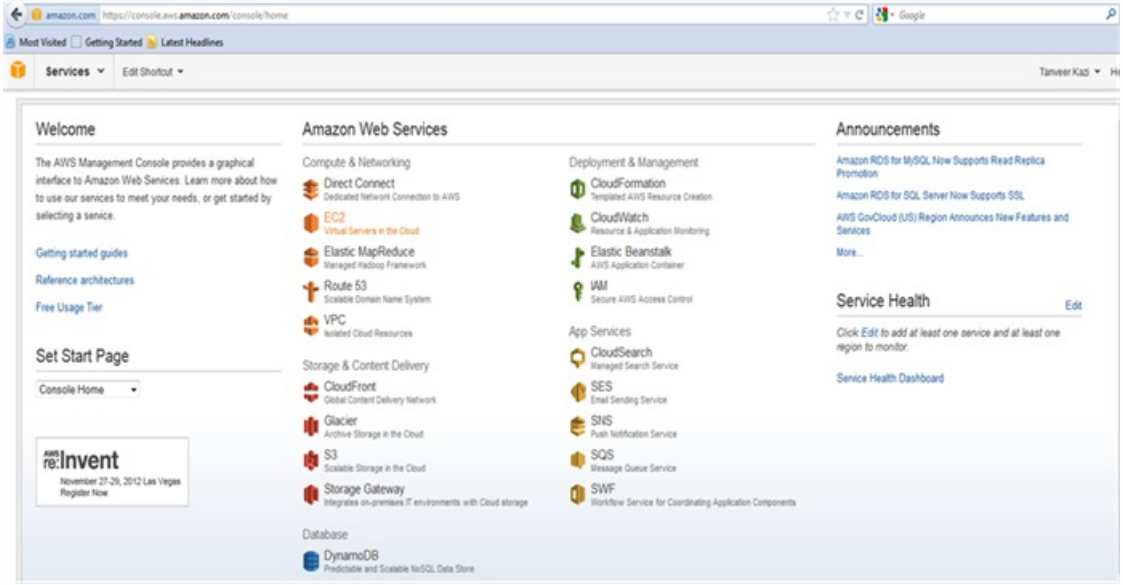
1. In a web browser, open the website at www.aws.amazon.com and log on with AWS credentials.



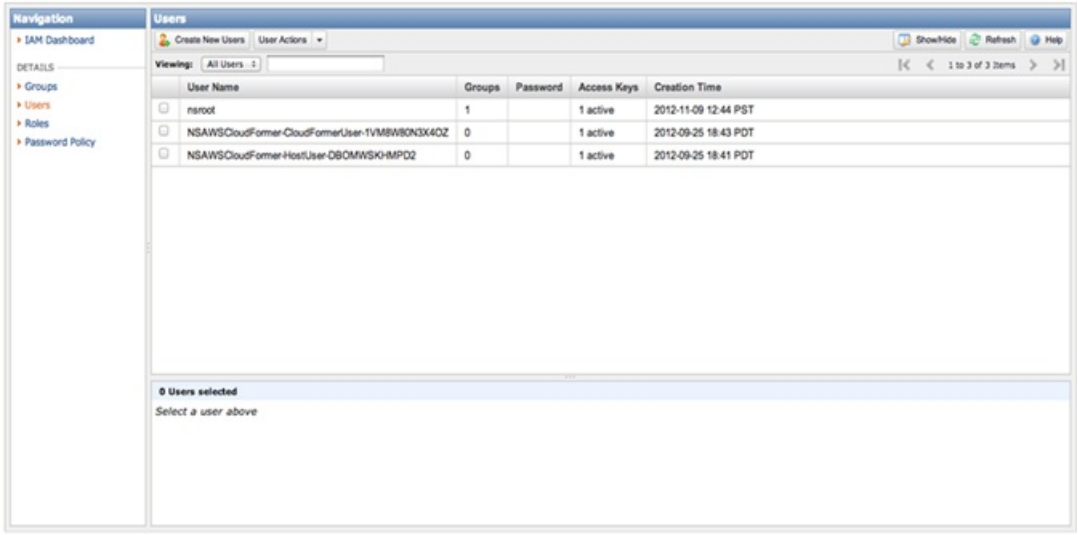
2. Click My Account/Console, and then click AWS Management Console.



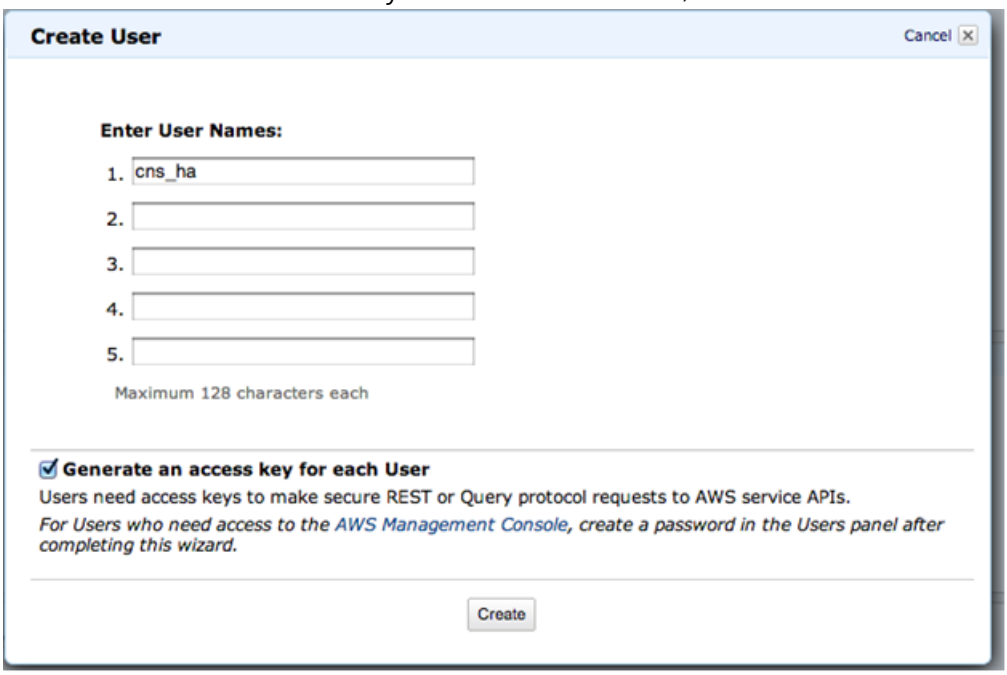
3. On the Amazon Web Services page, click IAM.



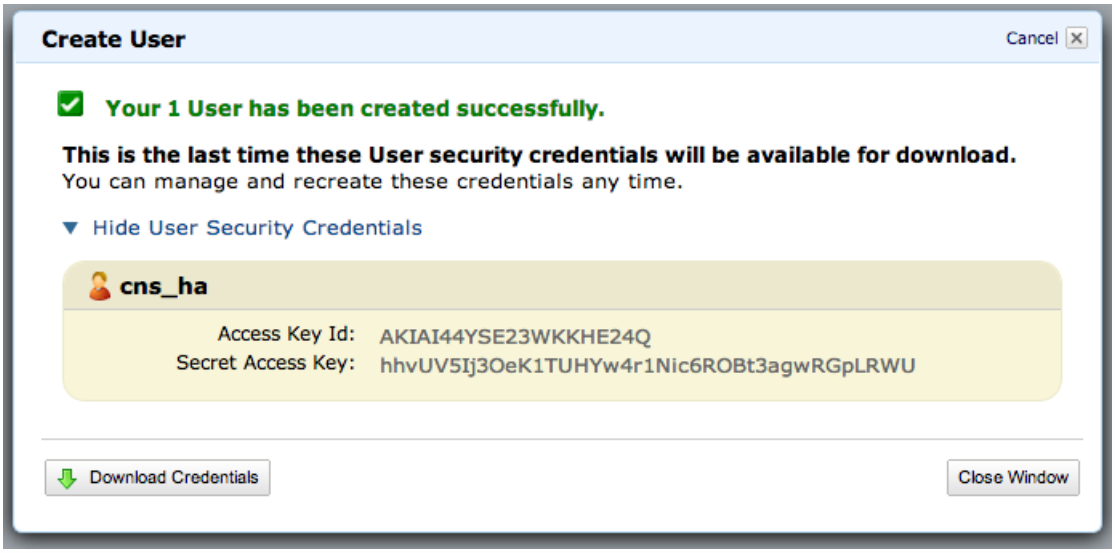
4. In the Navigation pane, click Users, and then click Create New Users.



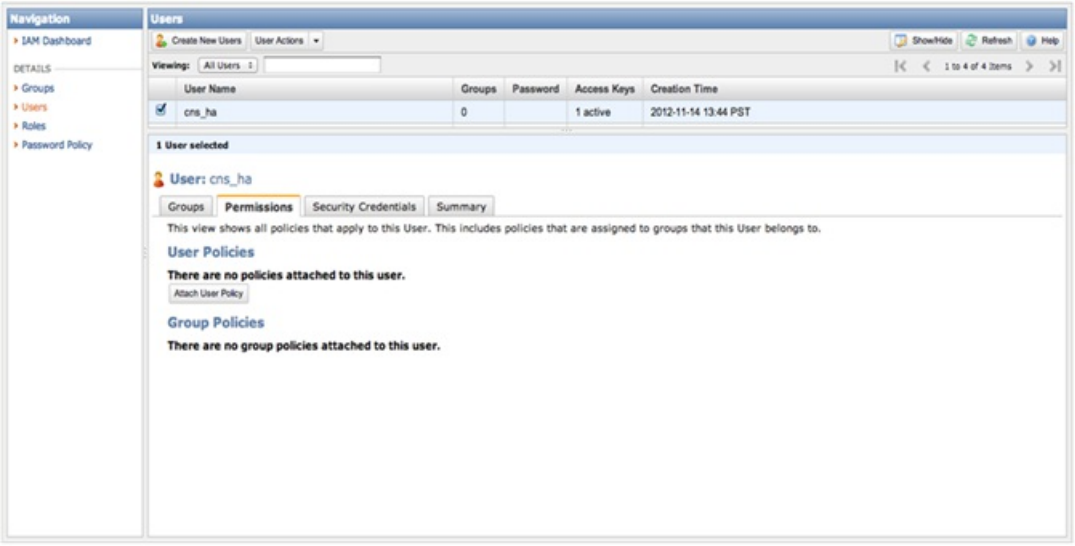
5. In the Create User dialog box, in one of the Enter User Names text boxes, type a user name (for example, cns_ha). Also select the Generate an access key for each User check box, and then click Create.



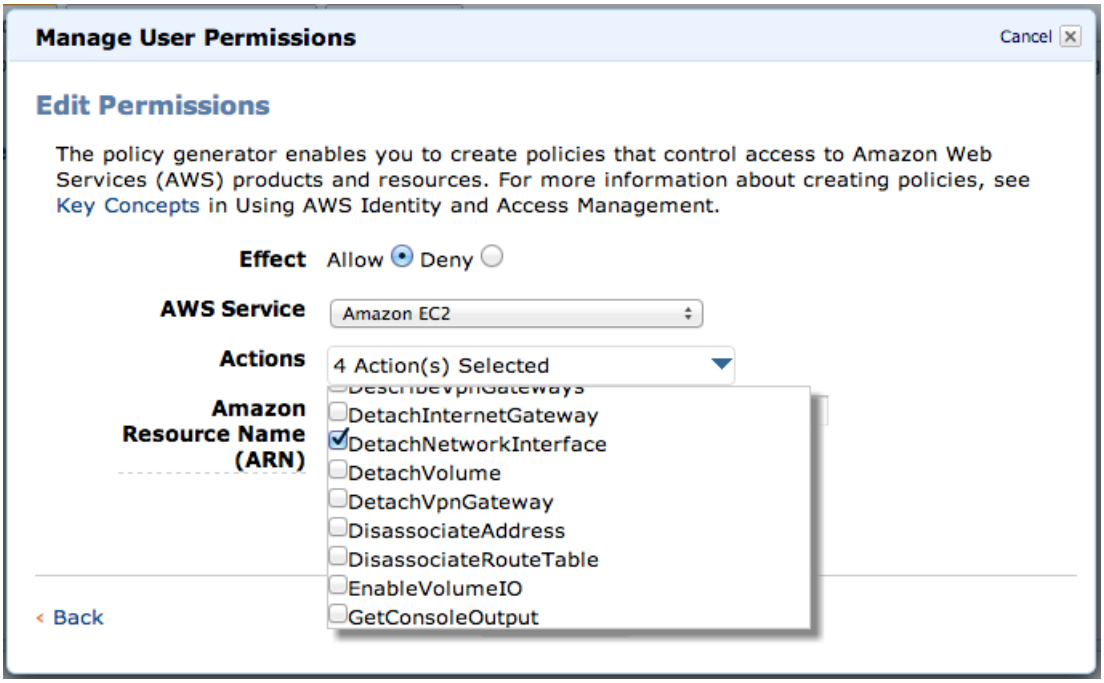
6. After a new IAM user is created, click Download Credentials to download the Access and Secret Keys to a safe location. These keys are required for launching NetScaler AMI. Click Close.
   Note: The Access Key ID and Secret Access Key values are used to create the key-pair file and to launch an instance.



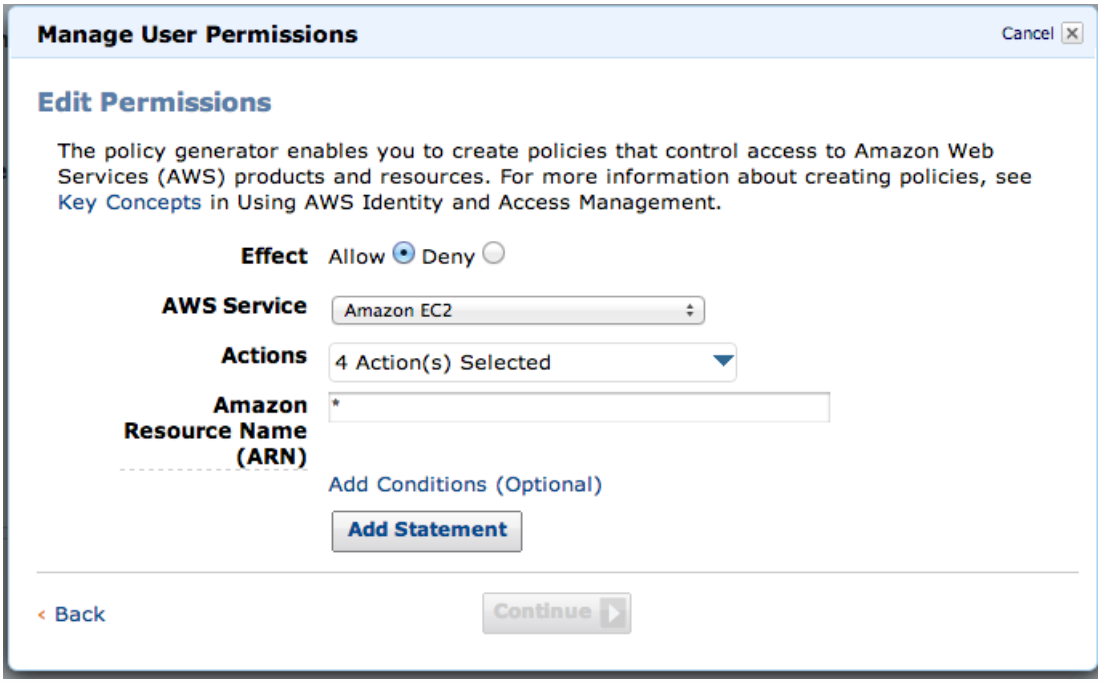
7. In the Users pane, select the newly created IAM user and click the Permissions tab. Then, click Attach User Policy to set policies for the user.



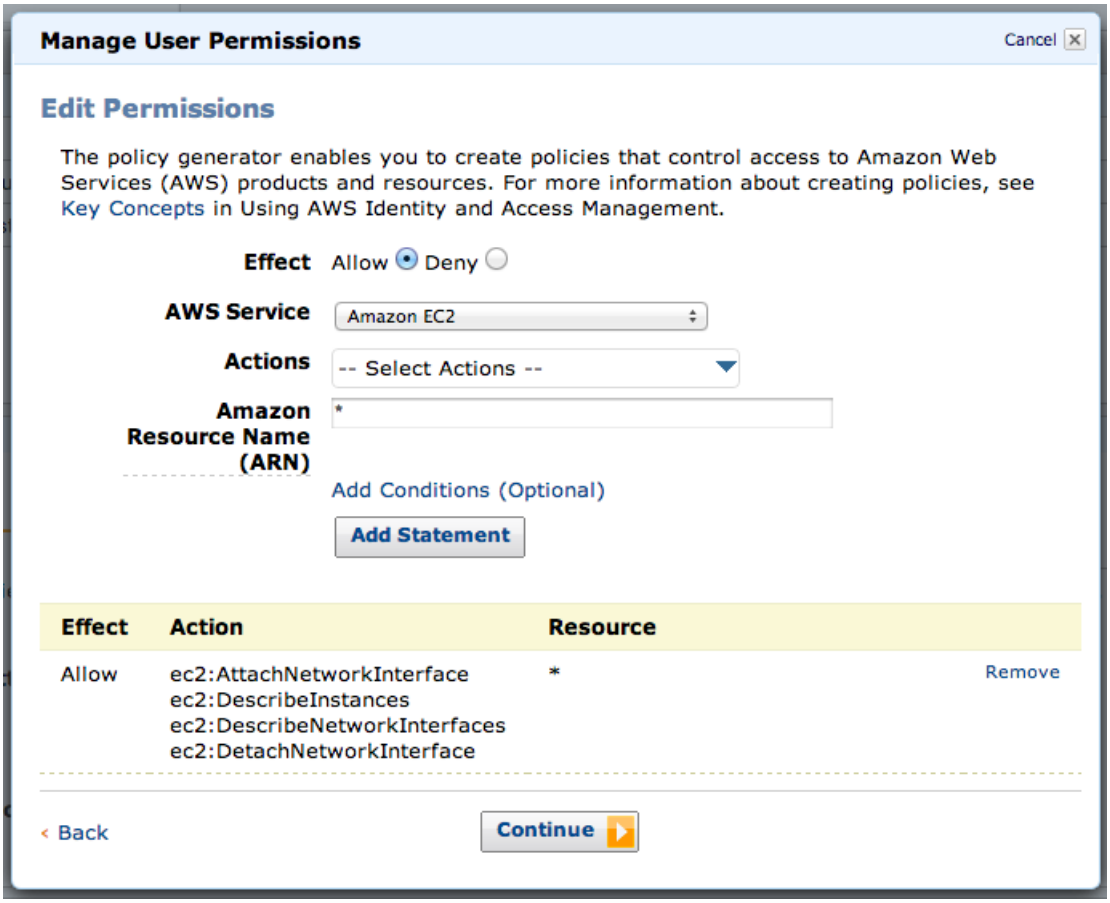
8. In the Manage User Permissions dialog box, next to Effect, select the Allow option. For AWS Service, select Amazon EC2. From the Actions drop-down list, select the following four actions:
   - AttachNetworkInterface
   - DescribeInstances
   - DescribeNetworkInterfaces
   - DetachNetworkInterface



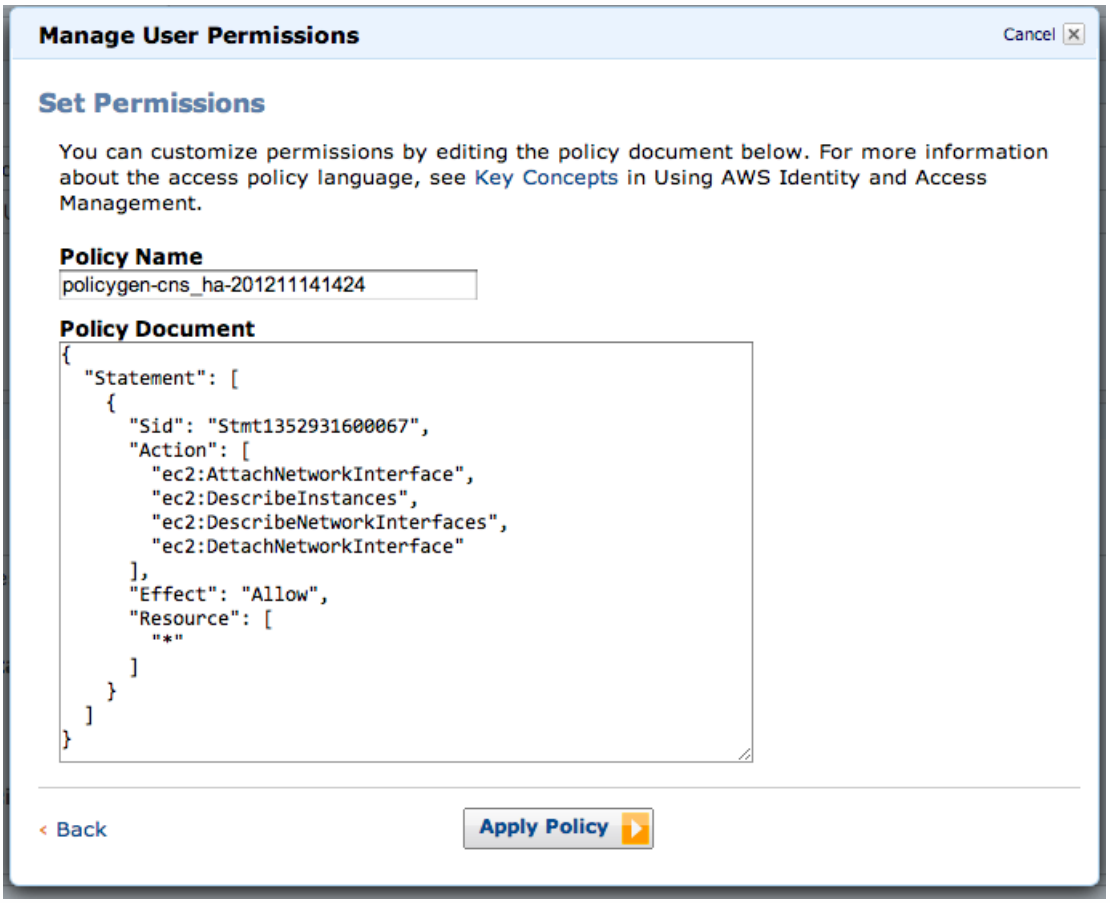
9. Click Add Statement.



10. Click Continue.



11. Click Apply Policy to set the new permissions for the selected user.



Launching the NetScaler AMI

Use the AWS CLI to launch the NetScaler AMI in an AWS VPC. Use the ec2-run-instances command. For information about the ec2-run-instances command, see http://docs.amazonwebservices.com/AWSEC2/latest/CommandLineReference/ApiReference-cmd-RunInstances.html.

Following are Windows and Linux examples of running the command to launch a single NetScaler instance. The EC2 instance type is m3.large. It is configured with the following entities:

- NetScaler AMI named ami-bd2986d4.
- Three ENIs (named NSIP, CLIENT-SIDE, and SERVER-SIDE) associated with the three subnets (15fa057e, 1547ba7e, and 1547ba7e) within the VPC.
- A single IP address for the NSIP ENI.
- Multiple private IP addresses (for multiple VIPs) on the CLIENT-SIDE ENI.
- Multiple private IPs (for multiple SNIPs) on the SERVER-SIDE ENI.

On a Windows platform:

C:\aws-vpc-config>ec2-run-instances ami-bd2986d4 -n 1 -t m1.large -k keyPairName -f access-secret-key-file -a :0:subnet-15fa057e:"NSIP":10.20.15.21 -a :1:subnet-1547ba7e:"CLIENT-SIDE":10.20.10.21::::"10.20.10.22,10.20.10.23,10.20.10.24,10.20.10.25,10.20.10.26,10.20.10.27,10.20.10.28,10.20.10.29,10.20.10.30" -a :2:subnet-cc47baa7:"SERVER-SIDE":10.20.1.21::::"10.20.1.22,10.20.1.23,10.20.1.24,10.20.1.25,10.20.1.26,10.20.1.27,10.20.1.28,10.20.1.29,10.20.1.30"
Note: The access-secret-key-file file contains the access and secret keys.

On a Linux platform:

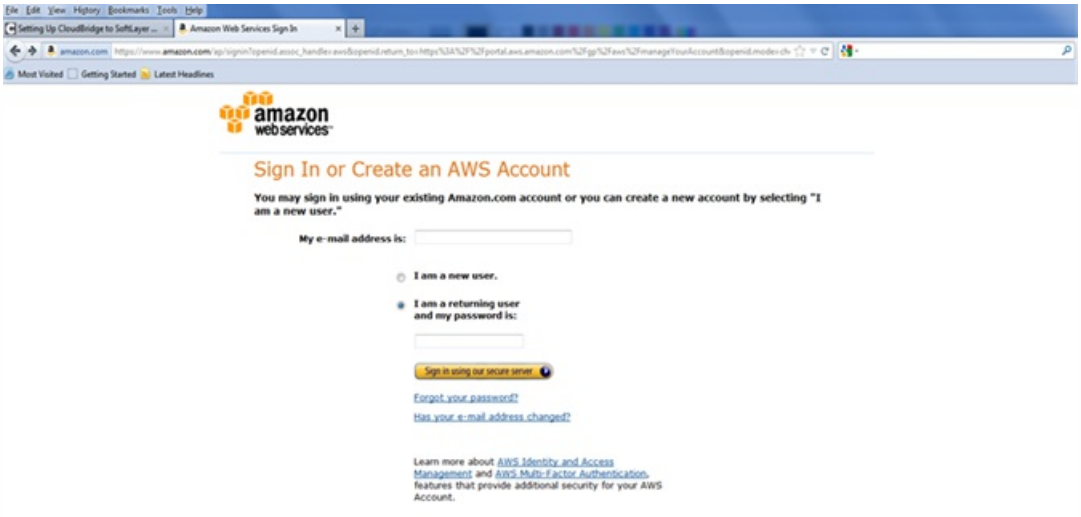
AWS PROMPT > ec2-run-instances ami-bd2986d4 -n 1 -t m1.large -k keyPairName -f access-secret-key-file -a :0:subnet-15fa057e:"NSIP":10.20.15.21 -a :1:subnet-1547ba7e:"CLIENT-SIDE":10.20.10.21::::10.20.10.22,10.20.10.23,10.20.10.24,10.20.10.25,10.20.10.26,10.20.10.27,10.20.10.28,10.20.10.29,10.20.10.30 -a :2:subnet-cc47baa7:"SERVER-SIDE":10.20.1.21::::10.20.1.22,10.20.1.23,10.20.1.24,10.20.1.25,10.20.1.26,10.20.1.27,10.20.1.28,10.20.1.29,10.20.1.30
Note: The access-secret-key-file file contains the access and secret keys.

The command returns the instance ID and the associated information. You can see the instance running within your AWS GUI Console.

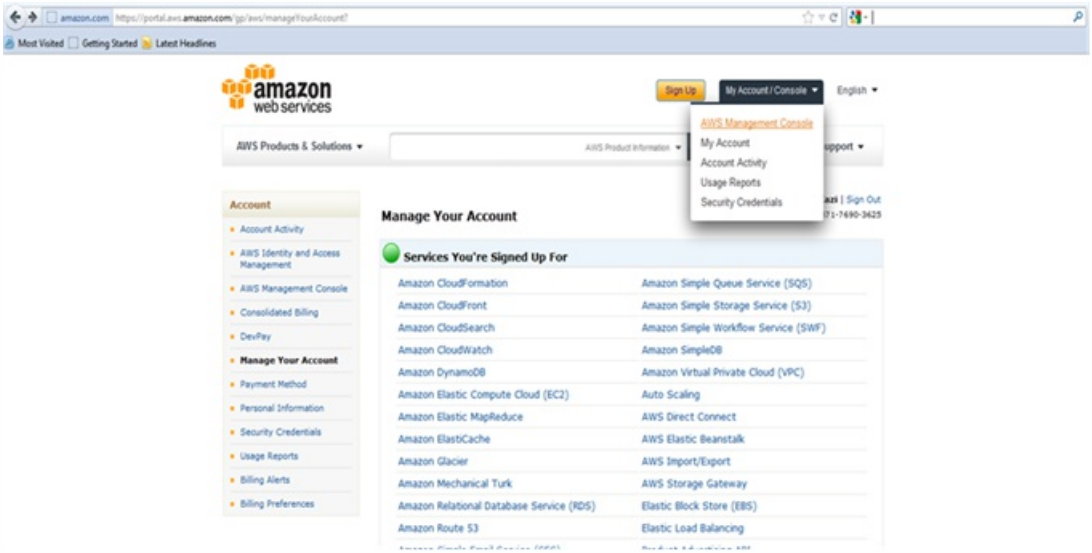
Note: Make sure that the environment variable EC2_URL points to the region where you want to launch the VPX instance.

To access the EC2 instance

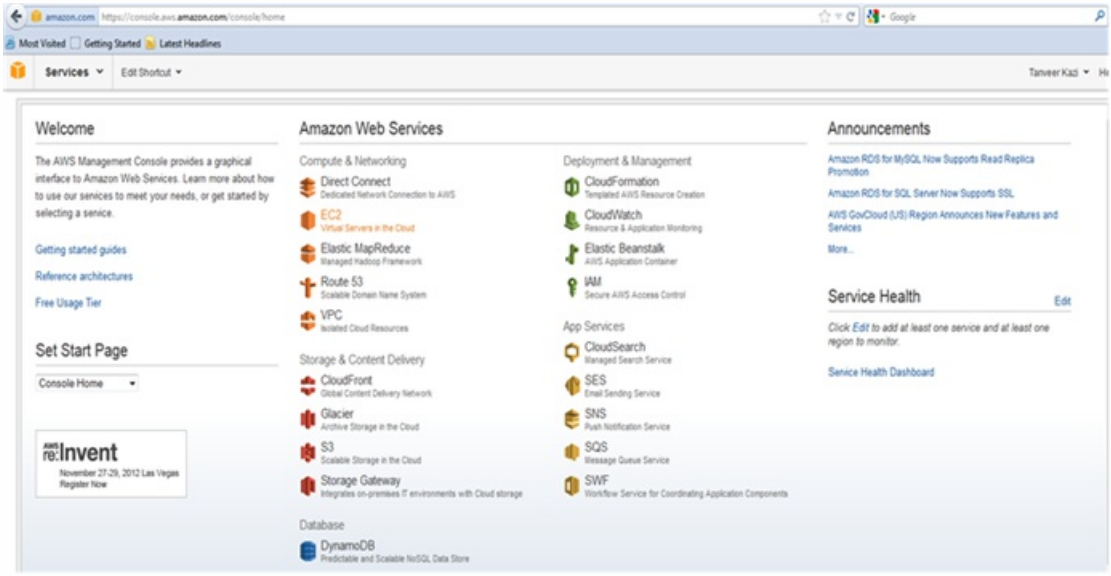
1. In a web browser, open the website at www.aws.amazon.com and log on with AWS credentials.



2. Click My Account/Console, and then click AWS Management Console.

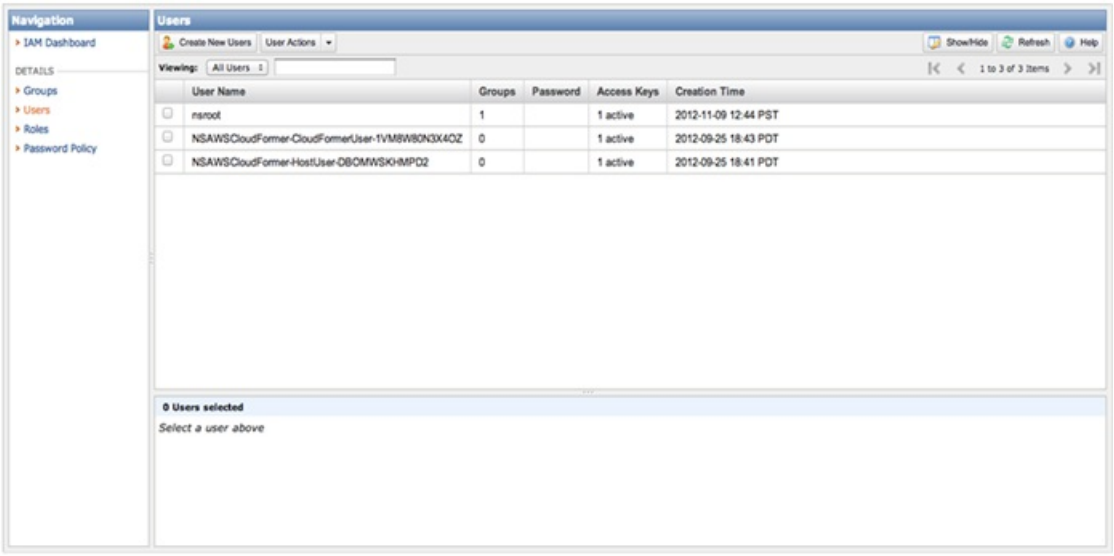


3. On the Amazon Web Services page, click EC2.



4. On the Amazon EC2 Console Dashboard page, in the Navigation pane, click Instances and verify that all of the NetScaler VPX instances are configured with the IP addresses that you specified when you used the ec2-run-instances command.

Note: The VPX instance or instances can take from five to ten minutes to start running.



The ec2-run-instances command does not allow associating AWS elastic IP with an ENI. To associate one or more EIPs with an ENI in the Navigation pane, in the NETWORK & SECURITY area, click Elastic IPs and associate EIPs with Private IP addresses for any of the VIPs that need to be externally routable.

You must also associate the instance ENIs with appropriate security groups. Go to the Network Interfaces section, right-click on the individual ENI, and select the Change Security Groups option. You can then associate a proper VPC security group.

Using the Citrix Cloud Formation Template to launch NetScaler VPX for AWS

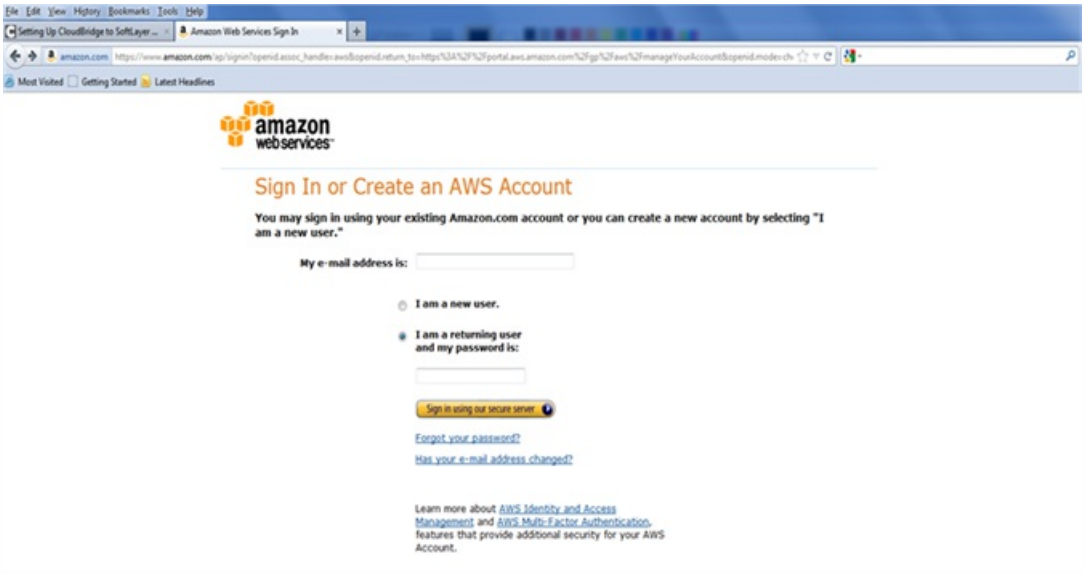
Citrix also provides a CloudFormation template that can be used to automate NetScaler instance launch. The tool requires an existing VPC environment. It launches a NetScaler instance with three ENIs. Therefore, to use the CloudFormation template, make sure that you have the following:

1. AWS account
2. AWS VPC
3. Three subnets within the VPC
4. A security group to use for the NetScaler instances ENIs

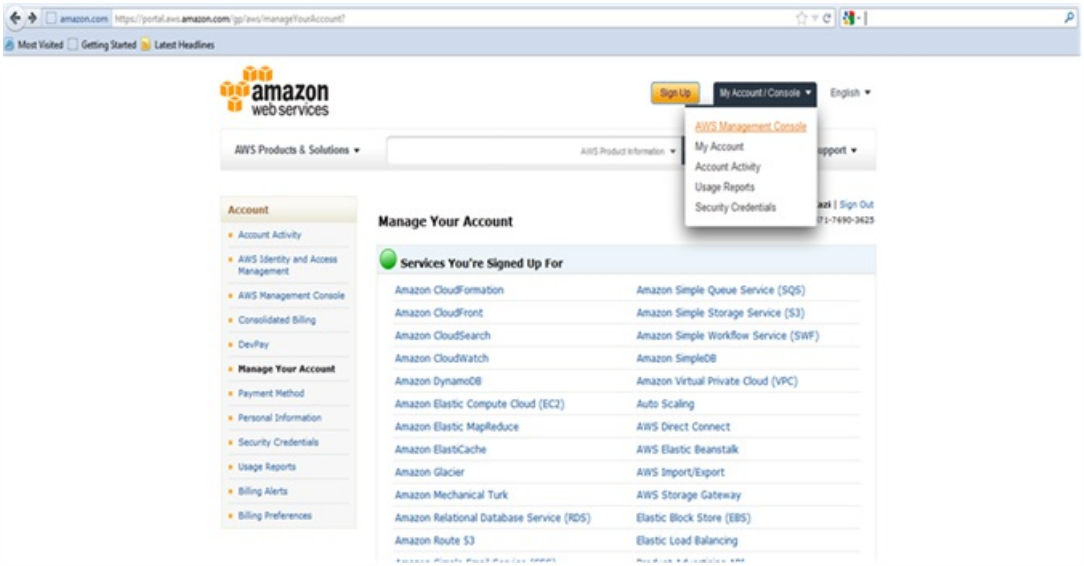
Refer to Creating an AWS Virtual Private Cloud (VPC) for information about how to configure subnets and security groups within a VPC. After configuring the required subnets and security groups, you can launch the NetScaler VPX AMI in AWS VPC. The CloudFormation tool provides functionality to launch a single NetScaler VPX instance or, to create a high availability environment, a pair of NetScaler VPX instances.

Launching a single NetScaler VPX instance in AWS

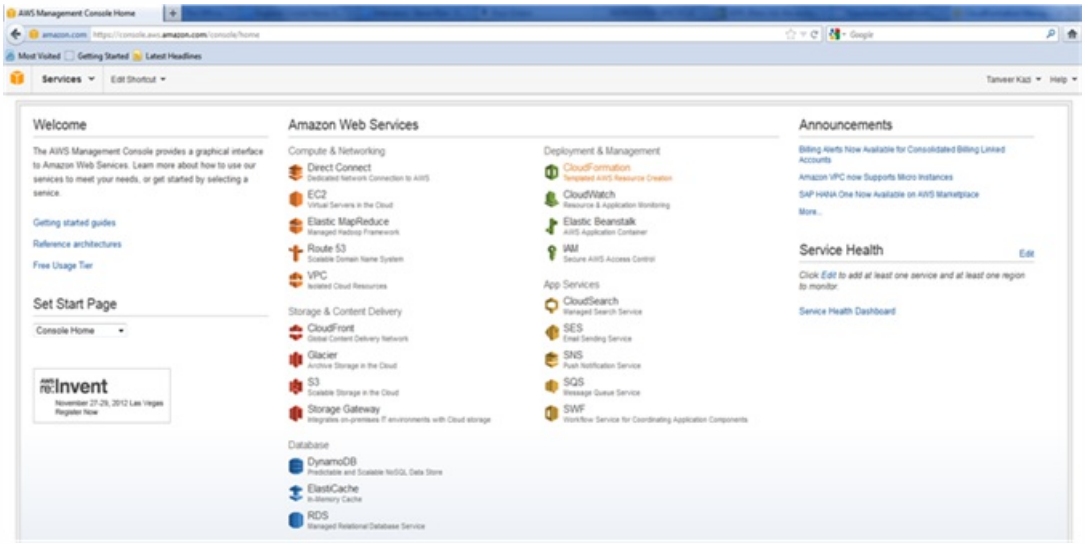
1. In a web browser, open the website at www.aws.amazon.comand log on with AWS credentials.



2. Click My Account/Console, and then click AWS Management Console.

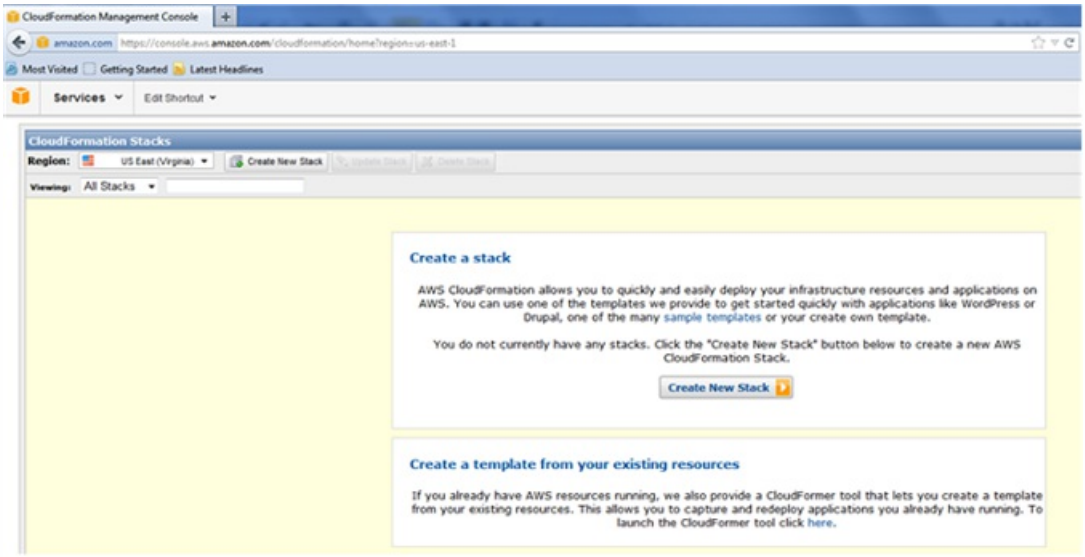


3. On the Amazon Web Services page, click Cloud Formation in the Deployment & Management section.



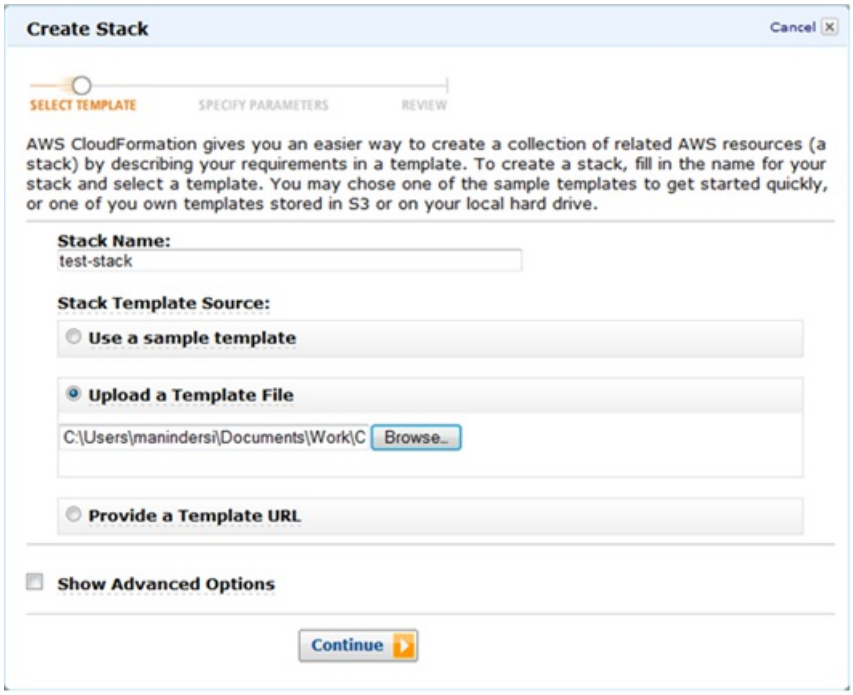
4. On the CloudFormation Stacks page, select the Region in which you plan to deploy the NetScaler VPX instance, and

then click Create New Stack.



5. In the Create Stack dialog box, specify a value for Stack Name, select the Upload a Template File option, and then click Browse. Select the template for a standalone NetScaler VPX from the local drive, and then click Continue.

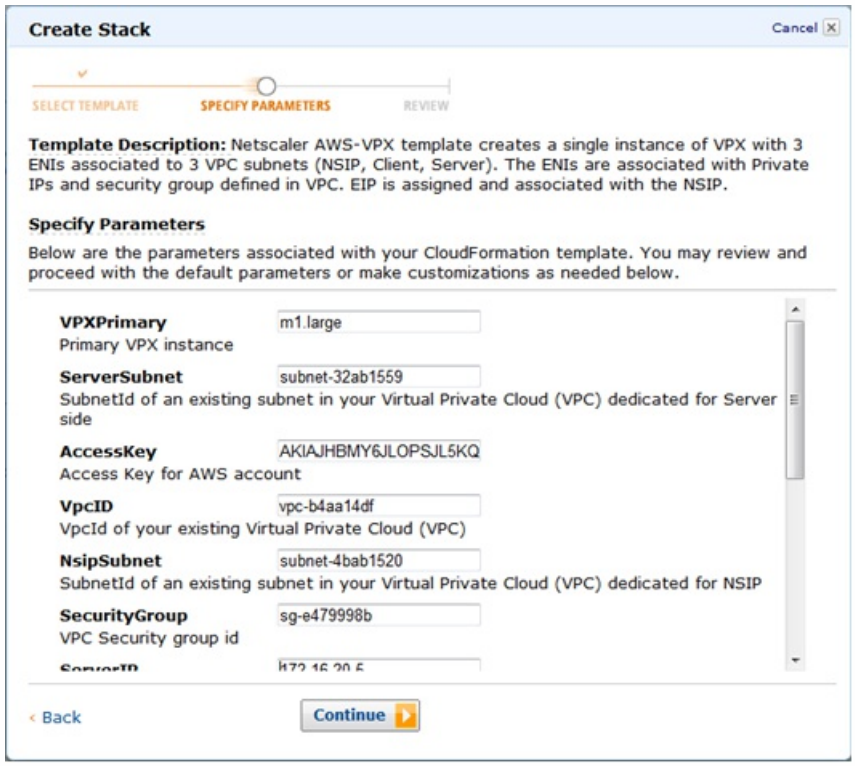
Note:



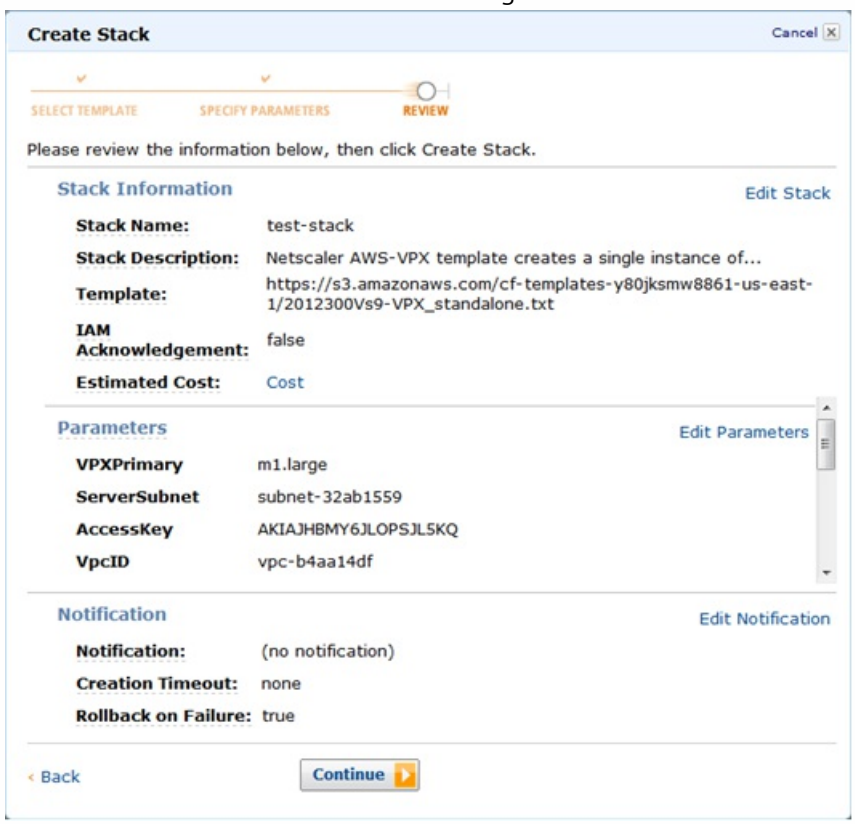
6. In the next pane, specify values for:
   - **VpcID** : An identifier to assign to the Virtual Private Cloud (VPC).
   - **NsipSubnet** : Subnet in which the NSIP is configured in the VPC
   - **ServerSubnet**: Subnet in which the server farm is configured in the VPC
   - **ClientSubnet**: SubnetId in which the client side is configured in the VPC
   - **SecurityGroup**: VPC Security group ID
   - **VPXPrimary**: Name of the primary VPX instance type
   - **AccessKey**: Access Key for IAM user account
   - **SecretKey**: Secret Key for IAM user account
   - **TenancyType**: Instance tenancy type, can be default or dedicated
   - **NsIP**: Private IP assigned to the NSIP ENI. The last octet of NSIP should be between 5 and 254.
   - **ServerIP**: Private IP assigned to the Server ENI. The last octet should be between 5 and 254.

- **ClientIP**: Private IP assigned to the Client ENI. The last octet should be between 5 and 254.
- **KeyName**: Name of an existing EC2 KeyPair to enable SSH access to the instances.

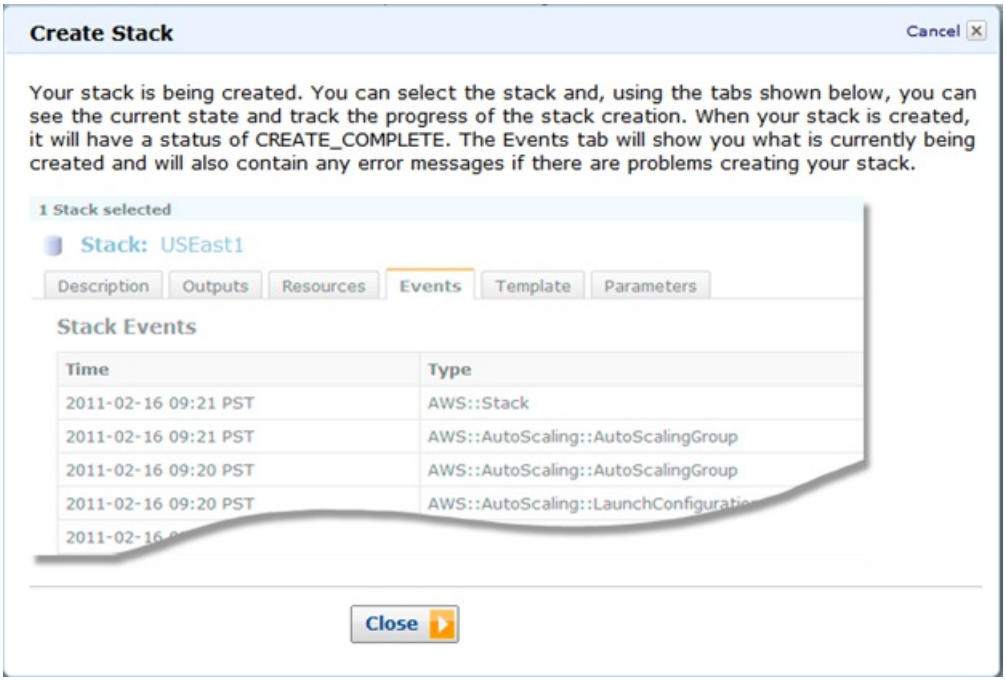
Note: Make sure that the VPC, subnets, security groups, routes and gateway associations are already configured.



7. Click Continue.
8. Review the values in the Create Stack dialog box.



9. Click Continue to create a Stack.



10. Click Close to close the Create Stack dialog box.
11. The new stack that you created appears on the CloudFormation Stacks page.

Note:

- Currently, the CloudFormation utility does not provide the functionality to add secondary IP addresses. Use the AWS console, after deploying a NetScaler VPX instance, to add the secondary IP addresses to the ENIs.
- The CloudFormation scripts for the standalone and HA pair VPX instances have the latest AMIs for the five supported regions. You have to update the scripts to synchronize with the latest AMIs.
- The script automatically selects the correct AMI for the region in which the VPX instance is being deployed.
- By default, all the ENIs are attached to one security group, use the AWS console to attach an ENI to a different security group.
- EIPs are automatically allocated and assigned to an instance. If the EIP limit exceeds the threshold for the region, the CloudFormation script fails and displays an error message.

Updated: 2015-01-29

1-Click helps you to launch an instance of NetScaler VPX on AWS, quickly as compared to other launching methods, with the default options. After the instance is launched on AWS, you can modify these options by using either the AWS CLI or the AWS GUI.

The default options include the following elastic network interfaces (ENIs) for the NetScaler instance:

- **Management Interface**—Associates a subnet for management related traffic. You add the NetScaler management IP (NSIP) address to this subnet.
- **Public Interface**—Associates a subnet for the client-access (user-to-NetScaler) traffic. You add one or more virtual IP (VIP) addresses on this subnet.
- **Private Interface**—Associates a subnet for server-access (NetScaler-to-server) traffic. You add subnet IP (SNIP) addresses on this subnet.

Before you begin launching an instance of NetScaler VPX on AWS, consider the following points :

- For security reasons, none of the elastic IP addresses are attached to the ENIs of the NetScaler VPX instance launched by using 1-Click. This means that the NetScaler VPX instance (including the management IP address) is not reachable from outside the AWS Virtual Private Cloud (VPC). If your VPC uses a Virtual Gateway or other method to provide a VPN access to the VPC, you can administer the instance by using the IP address of the network interface in the management subnet. If you do not have VPN access to your VPC, Citrix recommends that you set up a jump box instance within the VPC, and then use this as the source for accessing or managing other instances within the VPC. For instructions to create an SSH jump box, see https://s3.amazonaws.com/awsmp-usageinstructions/Creating_and_using_VPC.txt.
- Three default security policies are created. A policy each is attached to the management, public and private interfaces, respectively.
  - The security policy for the management interface allows traffic from a set of ports.
  - The security policies for the public and private interfaces block all the traffic to or from these interfaces. You can later modify these security groups to filter the desired traffic.
- High Availability configuration is not supported for a NetScaler VPX instance launched by using AWS 1-click.

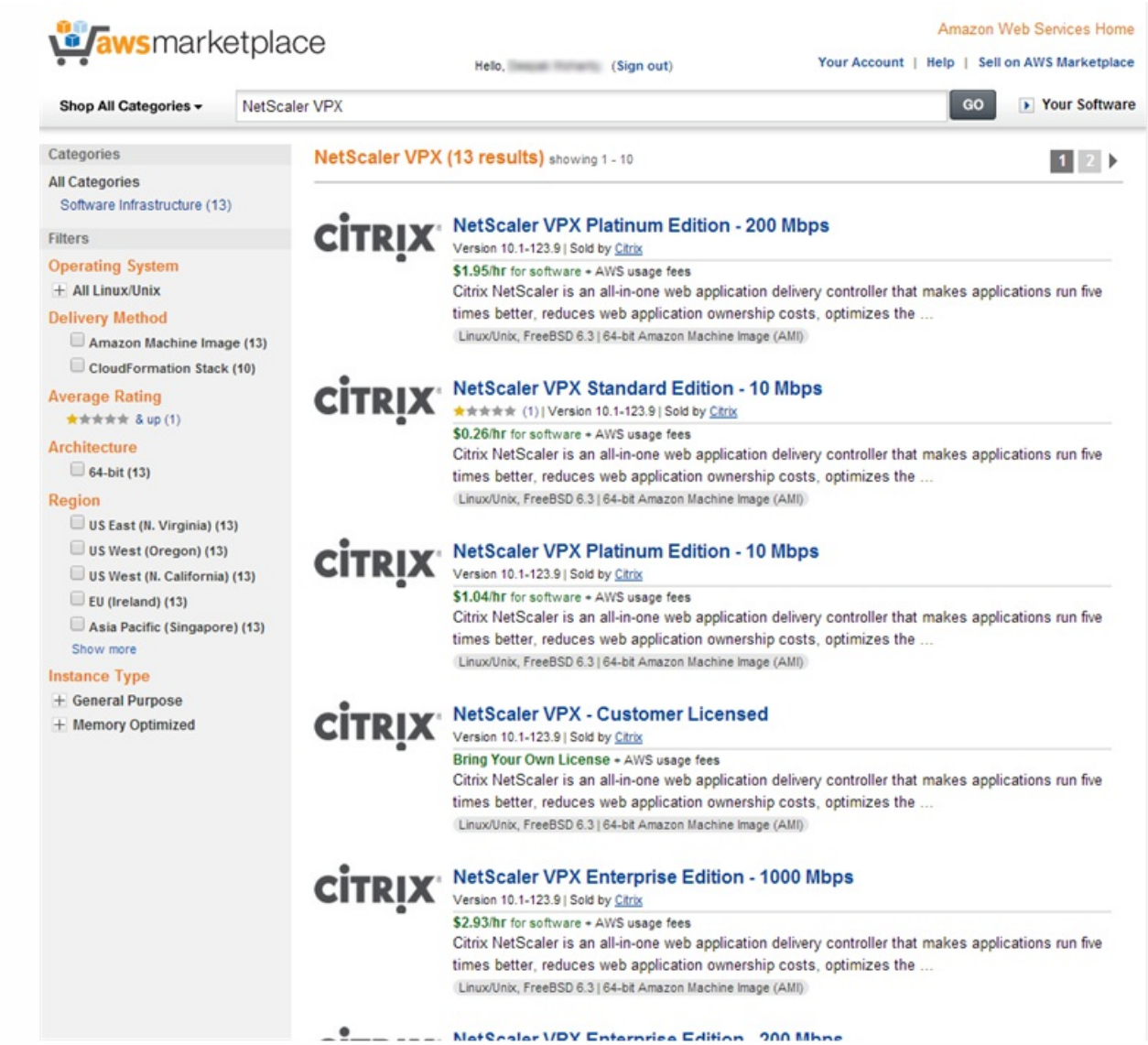
Before you begin launching an instance of NetScaler VPX on AWS, make sure that you have the following:

- An AWS account
- An AWS Virtual Private Cloud (VPC)
- Three subnets within the AWS VPC (one each for management interface, public interface, and private interface of the NetScaler instance)
- An IAM key pair

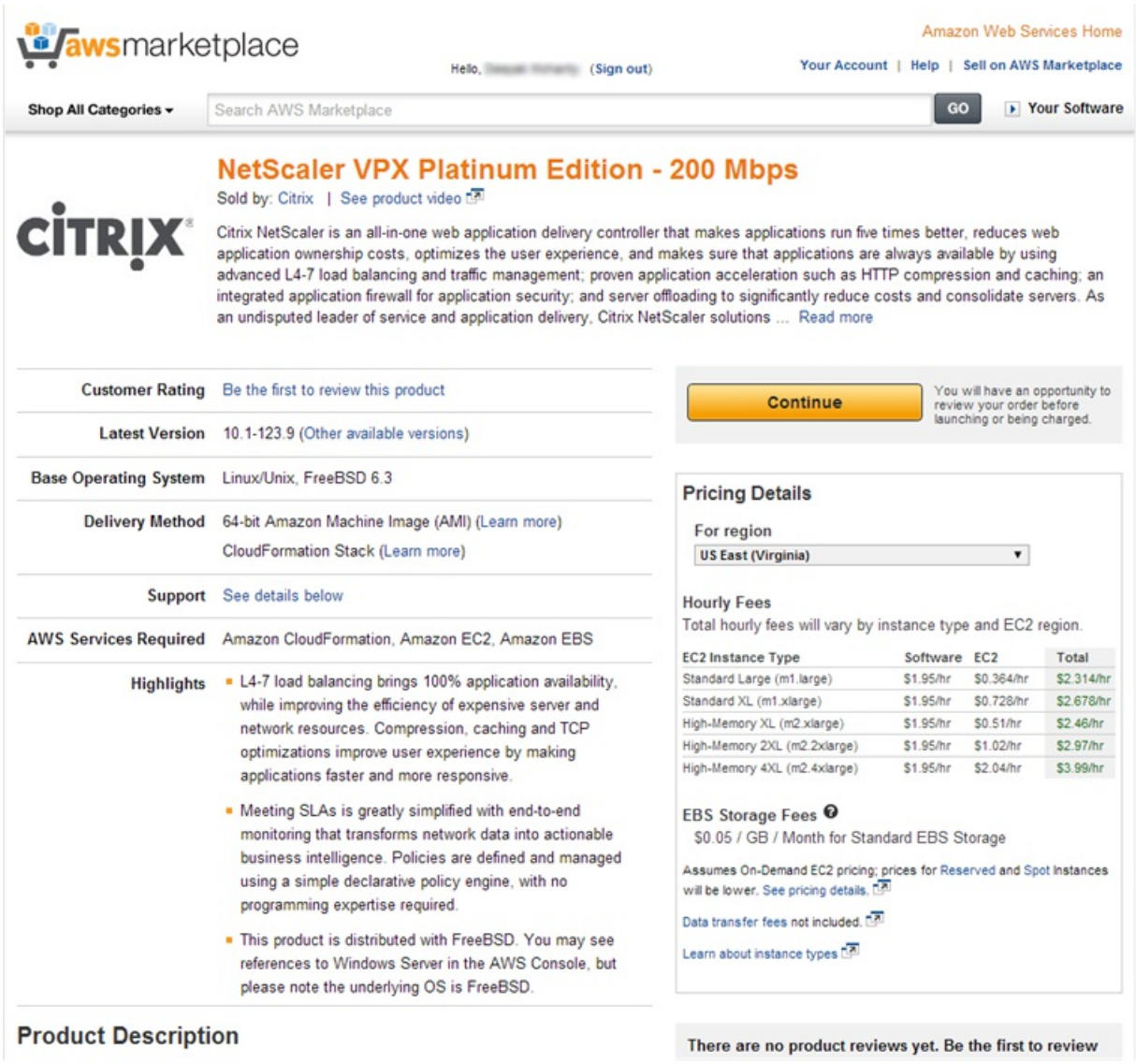
For information about creating an AWS account, a VPC, subnets in a VPC, and an IAM key pair, see Launching NetScaler VPX for AWS by Using the Amazon GUI and CLI toolkit.

To launch an instance of NetScaler VPX on AWS by using 1-Click

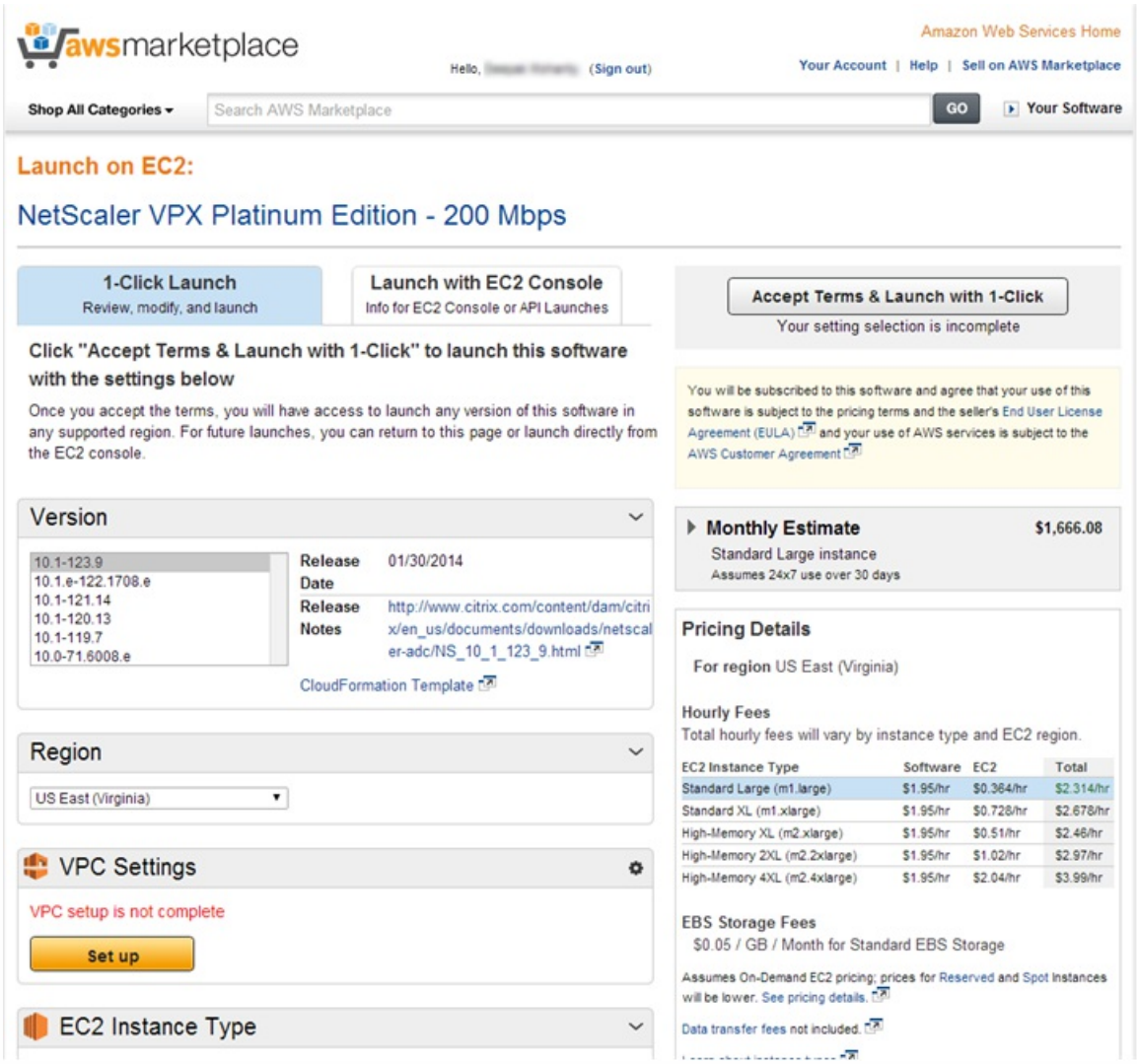
1. Log on to the AWS marketplace (https://aws.amazon.com/marketplace) by using your Amazon AWS credentials.
2. In the search field, type NetScaler VPX to search for the NetScaler AMI, and click Go.
3. On the search result page, click the desired Citrix NetScaler VPX offering.



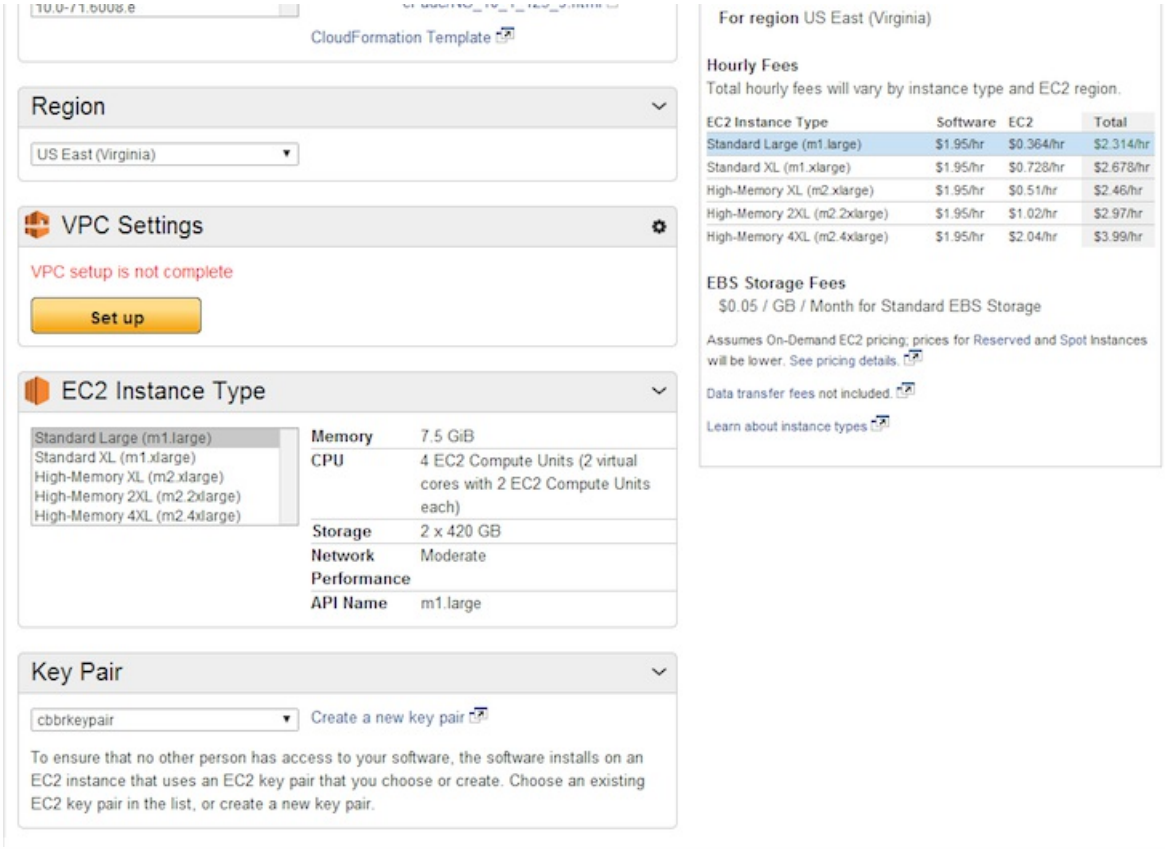
4. On the Citrix NetScaler VPX page, click Continue.



5. Click the 1-Click Launch tab. On the 1-Click Launch tab, specify values for the following fields:
   - Version
   - Region
   - EC2 Instance type
   - Key Pair

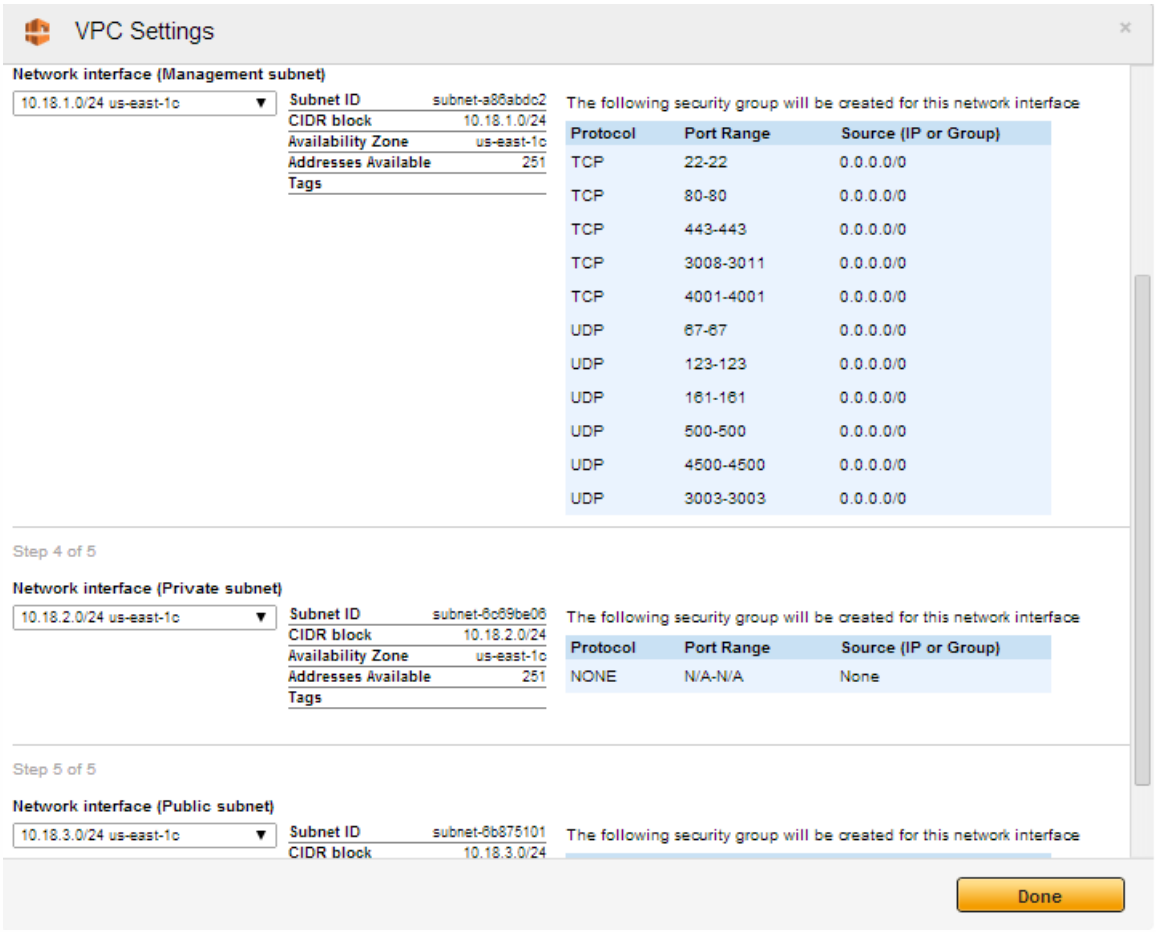


6. On the VPC Settings pane, click Setup.

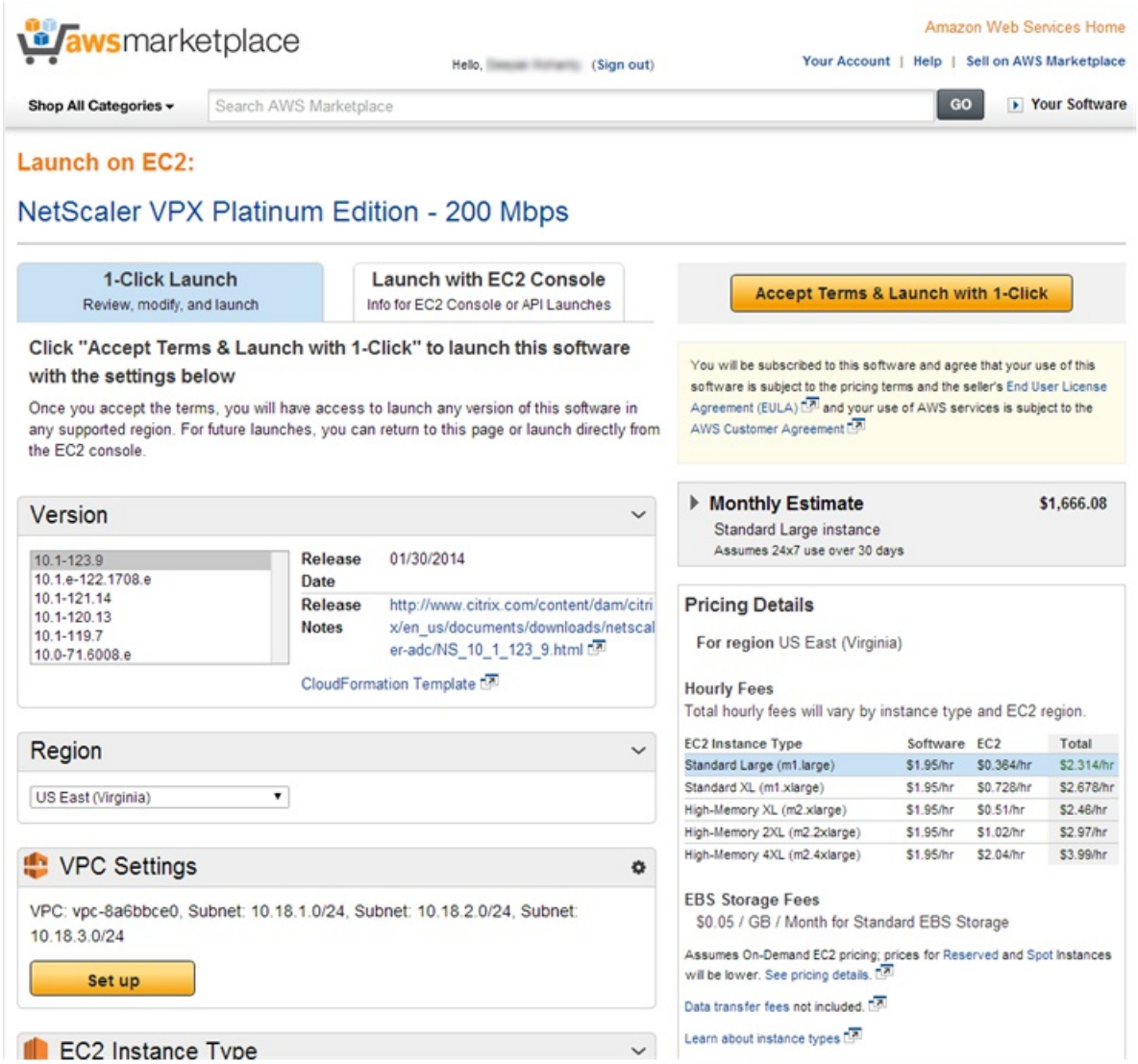


7. On the VPC Settings page, specify values for the following fields, and then click Done:
   - VPC
   - Network Interface (Management subnet)
   - Network Interface (Private subnet)
   - Network Interface (Public subnet)

   Note: You need to make sure that the subnets attached to these ENIs are different from each other. Attaching the same subnet to more than one ENI might cause routing issues.



8. Click Accept Terms & Launch with 1-Click.



After few minutes, the NetScaler instance is launched with three ENIs. You can now connect to the NSIP address (the IP address on the management ENI) of the instance by using the NetScaler CLI or NetScaler GUI and start configuring the NetScaler features, for example, load balancing.

Verifying the NetScaler VPX on AWS Installation

May 14, 2014

When the NetScaler instance is running, you can access the instance through the NetScaler GUI or the NetScaler CLI by connecting to the EIP associated with the management ENI (NSIP). For example, use the following addressing notation in a web browser:

http://<Elastic_IP> (unsecured access)

or

https://<Elastic_IP> (secured access)

Note:

- To access a NetScaler instance through SSH, provide the .pem file.
- You can use the AWS GUI console to manually add the private IP addresses for SNIPs on server subnets and VIPs on client subnets.
- If you want to access the NSIP from the Internet, you must assign an EIP to the NSIP address of each NetScaler instance. Also, make sure that the NSIP subnet is associated with a routing table that has a default route set to the Internet gateway.
- If you want VIP addresses to be accessible through the Internet, you must associate an EIP with each VIP address that is defined in the configuration.
- The following are the default administrator credentials to access a NetScaler VPX instance:
  - Username—nsroot
  - Password—The default password for the nsroot account is set to the AWS instance-ID of the NetScaler VPX instance. For a high availability configuration between two NetScaler VPX instances, the nsroot password of the secondary node is set to that of the primary node after the HA configuration synchronization.
- You can find the private key file from the AWS console. To view the private key file:
  1. Log on to the AWS marketplace (https://aws.amazon.com/marketplace) by using your Amazon AWS credentials.
  2. Click **Amazon Web Services Home**.
  3. Click **My Account/Console**, and then click **Security Credentials**.

Attaching Additional IP Addresses to an Instance

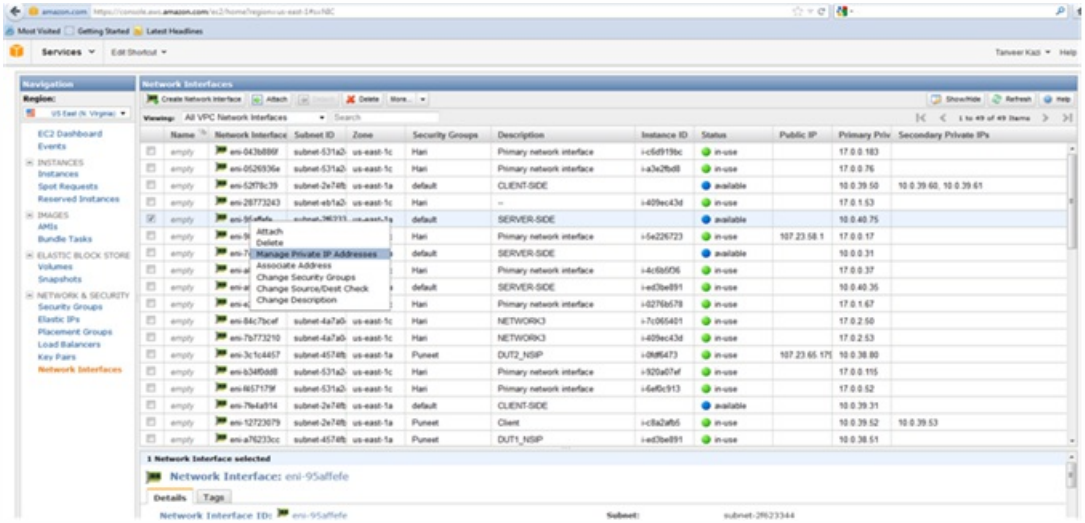
Aug 23, 2013

You can attach additional IP addresses to an instance as follows:

1. Add a secondary IP address to an ENI.
2. Associate an EIP with the secondary IP address that you created.

To add a secondary IP address to the ENI

1. In a web browser, open the website at www.aws.amazon.com and log on with AWS credentials.
2. Click My Account/Console, and then click AWS Management Console.
3. On the Amazon Web Services page, click EC2.
4. On the Amazon EC2 Console's Dashboard page, in the Navigation pane, in NETWORK & SECURITY, click Network Interfaces.
5. In the Network Interfaces pane, right-click the ENI attached to the subnet, and then select the Manage Private IP Addresses option from the pop-up menu.



6. In the Manage Private IP Addresses dialog box, click Assign a secondary private IP address and either let AWS automatically assign an IP address or type an IP address in the auto-assign text-field. Click Yes, Update.

Associating an EIP with the secondary IP

Complete the following steps to associate an EIP with a secondary IP address:

1. On the Amazon EC2 Console Dashboard page, in the Navigation pane, in NETWORK & SECURITY, click Elastic IPs.
2. In the Addresses pane, click Allocate New Address.
3. In the Allocate New Address dialog box, select VPC from the EIP used in drop-down list and click Yes, Allocate.
4. Select the newly allocated EIP, and click Associate Address.
5. In the Associate Address dialog box, select, from the **Instance** and the **Private IP address** drop-down lists, the instance and private address that you want to associate with the EIP. Then, click Yes, Associate.

Downloading a NetScaler VPX License

Jan 31, 2011

After the initial instance launch, NetScaler VPX for AWS requires a license. If you are bringing your own license (BYOL), see the *VPX Licensing Guide* at http://support.citrix.com/article/CTX122426

You have to:

1. Use the licensing portal within MyCitrix to generate a valid license.
2. Upload the license to the instance.

If this is a **paid** marketplace instance, then you do not need to install a license. The correct feature set and performance will activate automatically.

Load Balancing Servers in different Availability Zones

Nov 27, 2012

A NetScaler instance can be used to load balance servers running in the same availability zone, or in:

- A different availability zone (AZ) in the same AWS VPC
- A different AWS region
- AWS EC2 in a VPC

To enable NetScaler to load balance servers running outside the AWS VPC that the NetScaler instance is in, configure the NetScaler to use EIPs to route traffic through the Internet gateway, as follows:

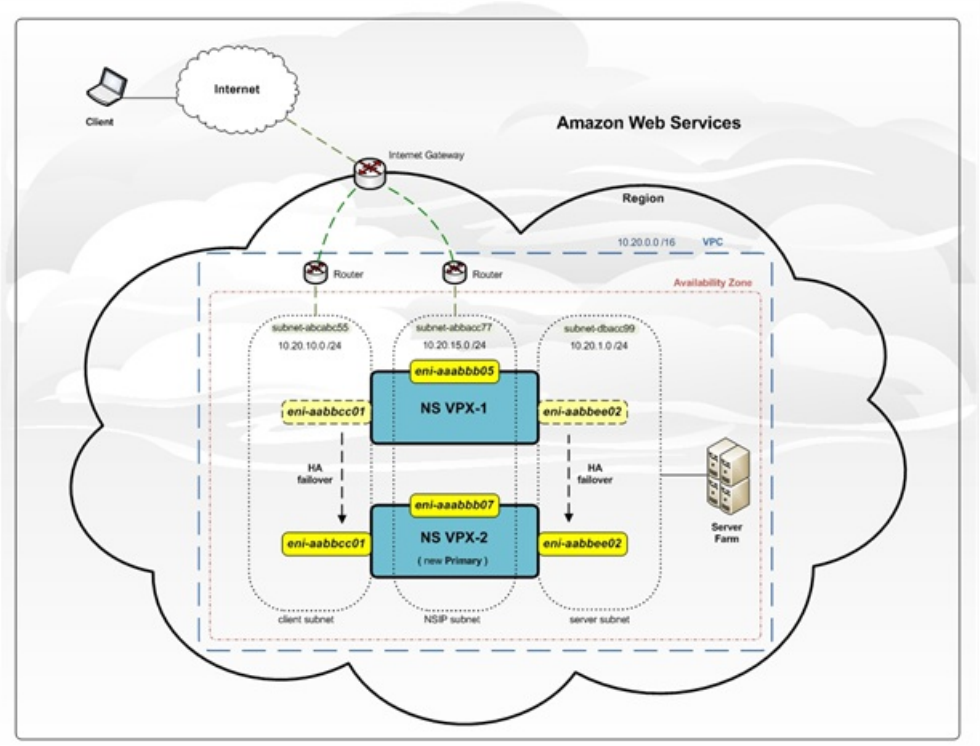
1. Configure a SNIP on the NetScaler by using the NetScaler CLI or the NetScaler GUI
2. Enable traffic to be routed out of the AZ, by creating a public facing subnet for the server-side traffic.
3. Add an Internet gateway route to the routing table, using the AWS GUI console.
4. Associate the routing table you just updated with the server-side subnet.
5. Associate an EIP with the server-side private IP address that is mapped to a NetScaler SNIP address.

High Availability

May 13, 2014

Two Citrix® NetScaler® VPX™ instances in AWS can be configured as a high availability (HA) pair. With one instance configured as the primary node and the other as the secondary node, the primary node accepts connections and manages servers while the secondary node monitors the primary. If, for any reason, the primary node is unable to accept connections, the secondary node takes over.

The following figure shows an example of the HA deployment architecture for NetScaler VPX on AWS.
Figure 1. NetScaler VPX on AWS - HA Deployment



To deploy HA for VPX on AWS, you must configure at least two ENIs on the primary instance and a single ENI on the secondary instance. On each instance, configure the NetScaler IP (NSIP) address (the management address) on the default ENI. On the primary instance, use the additional ENIs for client and server connections.

For instructions on obtaining access and secret keys, in the AWS documentation, see "How Do I Get Security Credentials?" and "Creating, Modifying, and Viewing User Access Keys (AWS Management Console)." For instructions to create an IAM user and set permissions, see "Creating an IAM Account."

Example format for a key file is:
ACCESS_KEY="AKIAJPBBBBBBBVA2PR2OHJNA"
SECRET_KEY="d75KxU7ukd44444NNNNtrrAOgynwBdJoSiooP"
Note: For HA failover to work:

1. The NSIP addresses for each NetScaler instance in an HA pair must be configured on the default ENI of the instance.
2. Both the primary and secondary instances must have EIPs associated with the NSIP or NAT configured to handle outgoing traffic in order to have access to the AWS API servers.
3. Client and server traffic (data-plane traffic) must not be configured on the default ENI.

4. Access and secret keys associated with the user's AWS Identity and Access Management (IAM) account. If the correct key information is not used when creating VPX instances, the HA deployment will fail. The access and secret keys are required for sending Query APIs to the AWS server.
5. Nameservers/DNS servers are configured at VPC level using DHCP options.

Notes on HA:

- Because Amazon does not allow any broadcast/multicast packets in AWS, HA is implemented by migrating data-plane ENIs from the primary to the secondary (new primary) VPX instance when the primary VPX instance fails.
- To deploy HA for VPX on AWS, you must configure at least two ENIs on the primary instance and a single ENI on the secondary instance.
- Because the default ENI cannot be moved to another VPX instance, you should not use the default ENI for data.
- The message AWSCONFIG IOCTL NSAPI_HOTPLUG_INTF success output 0 indicates that the two data ENI's have successfully attached to the secondary instance (the new primary).
- Failover might take up to 20 seconds due to the AWS detach/attach ENI mechanism.
- Upon failover, the failed instance always restarts.
- The secondary node always has one ENI interface (for management) and the primary node can have up to four ENIs.
- The heartbeat packets are received only on the management interface.
- The configuration file of the primary and secondary NetScaler appliances is synchronized, including the nsroot password. The nsroot password of the secondary node is set to that of the primary node after the HA configuration synchronization.
- **The AWS debug messages are available in the log file, /var/log/ns.log, on the VPX instance**.

To deploy HA for two VPX instances on AWS, you must create the primary NetScaler VPX instances with three ENIs and the secondary NetScaler VPX with a single ENI.

Following is an example of launching a primary VPX instance with three ENIs:

C:\aws-vpc-config>ec2-run-instances ami-bd2986d4 -n 1 -t m1.large -k keyPairName -f ./access-secret-key-file -a :0:subnet-15fa057e:"NSIP":10.20.15.21 -a :1:subnet-1547ba7e:"CLIENT-SIDE":10.20.10.21::::"10.20.10.22,10.20.10.23,10.20.10.24,10.20.10.25,10.20.10.26,10.20.10.27,10.20.10.28,10.20.10.29,10.20.10.30" -a :2:subnet-cc47baa7:"SERVER-SIDE":10.20.1.21::::"10.20.1.22,10.20.1.23,10.20.1.24,10.20.1.25,10.20.1.26,10.20.1.27,10.20.1.28,10.20.1.29,10.20.1.30"
Following is an example of launching a secondary VPX instance with a single ENI:

C:\aws-vpc-config>ec2-run-instances ami-bd2986d4 -n 1 -t m1.large -k keyPairName -f access-secret-key-file -a :0:subnet-15fa057e:"NSIP":10.20.15.31
Note: The access-secret-key-file argument contains the access and secret key. (You cannot change the access-secret-key-file associated with a VPC instance after it is created.)
After the two NetScaler instances are UP, configure the HA pairing on both the instances. You have to configure the instance with two or more ENIs before configuring HA on the instance with one ENI. Use the add HA node command, from within the NetScaler CLI, or from the NetScaler GUI. For example:

On the VPX instance with two or more ENIs:

add HA node 1 10.20.15.31

On the VPX instance with one ENI:

add HA node 1 10.20.15.21

After you enter add HA node commands, the two nodes form an HA pair, and configuration information is synchronized between the two VPX instances.

To remove HA from NetScaler VPX pair

You can remove HA configuration from the NetScaler VPX pair by using the remove ha node command. You have to remove the HA configuration from the secondary NetScaler VPX before removing the HA configuration from the primary NetScaler VPX.
For example, on the Secondary NetScaler VPX instance, at the NetScaler command line, type:

remove ha node

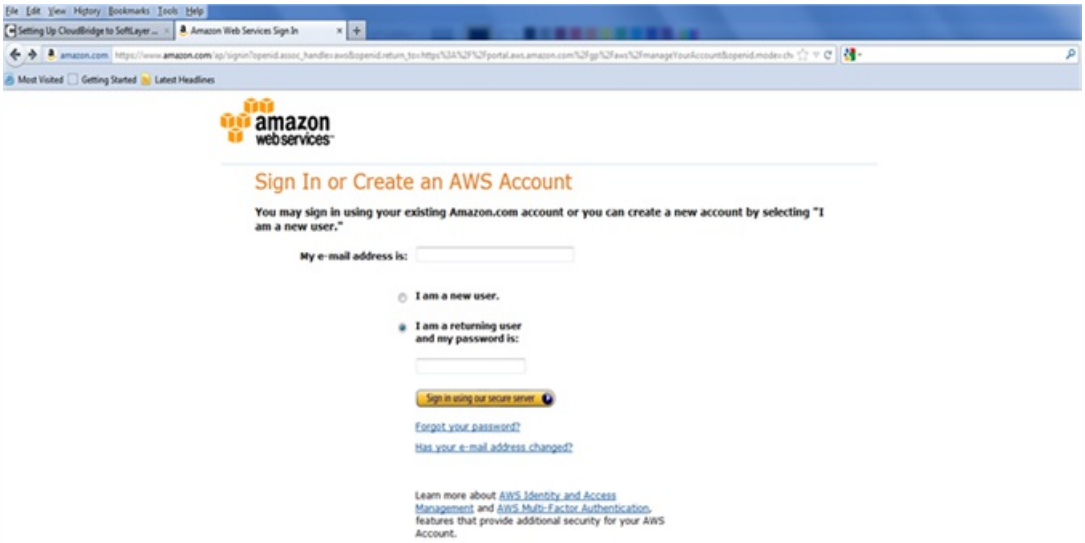
save config

On the Primary NetScaler VPX instance, at the NetScaler command line, type:

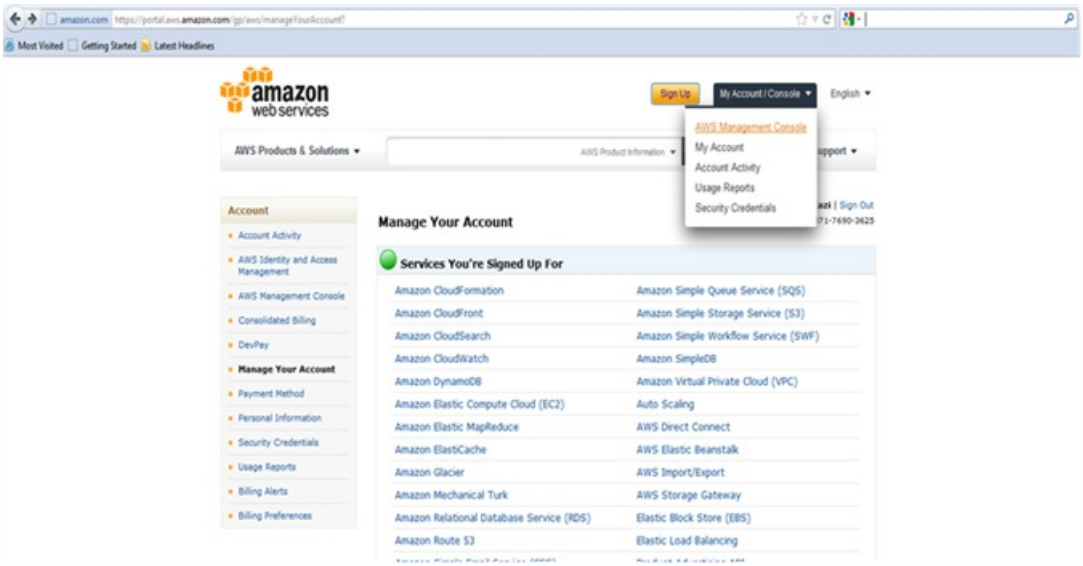
remove ha node

save config

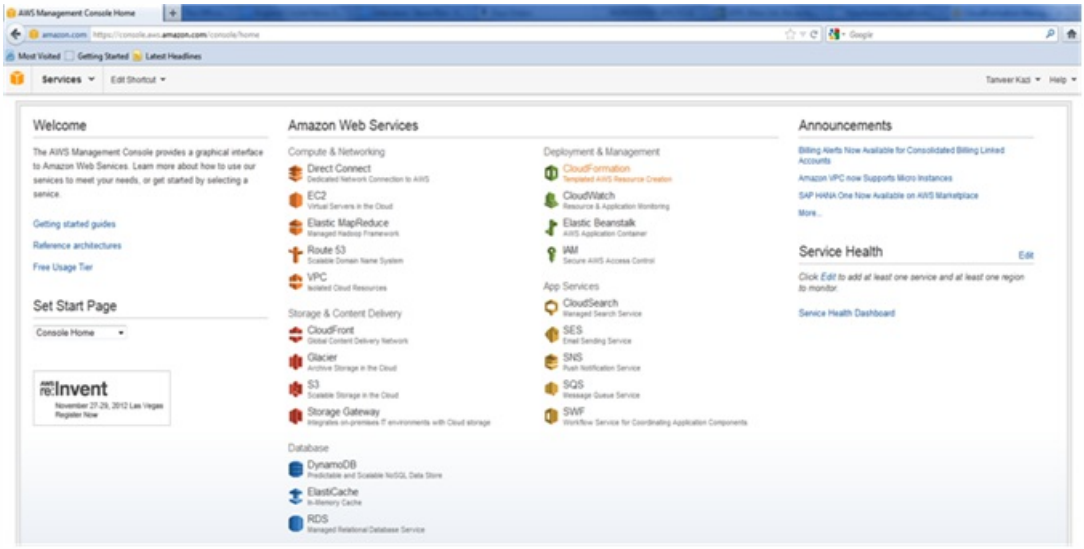
1. In a web browser, open the website at www.aws.amazon.com and log on with AWS credentials.



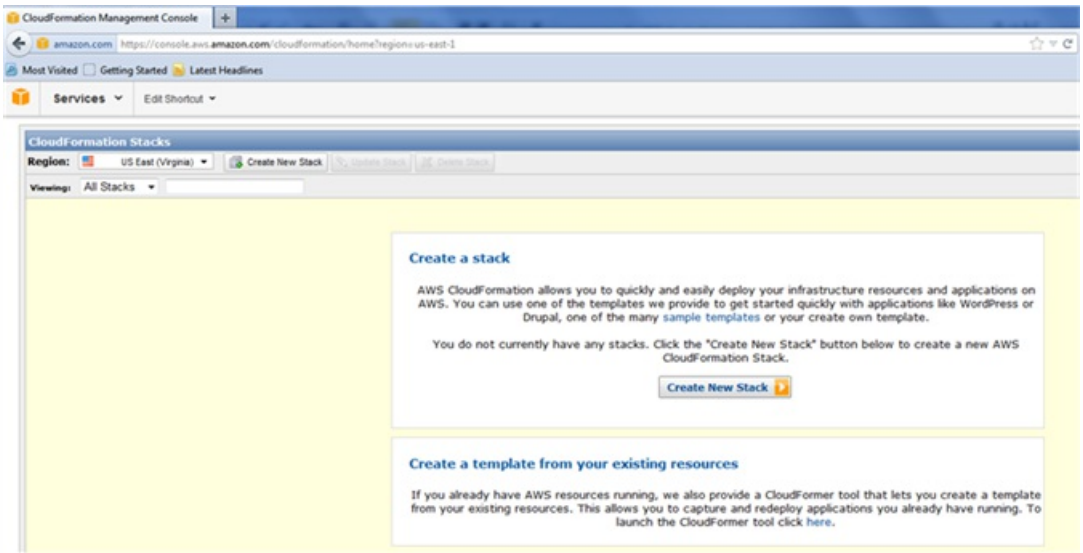
2. Click My Account/Console, and then click AWS Management Console.



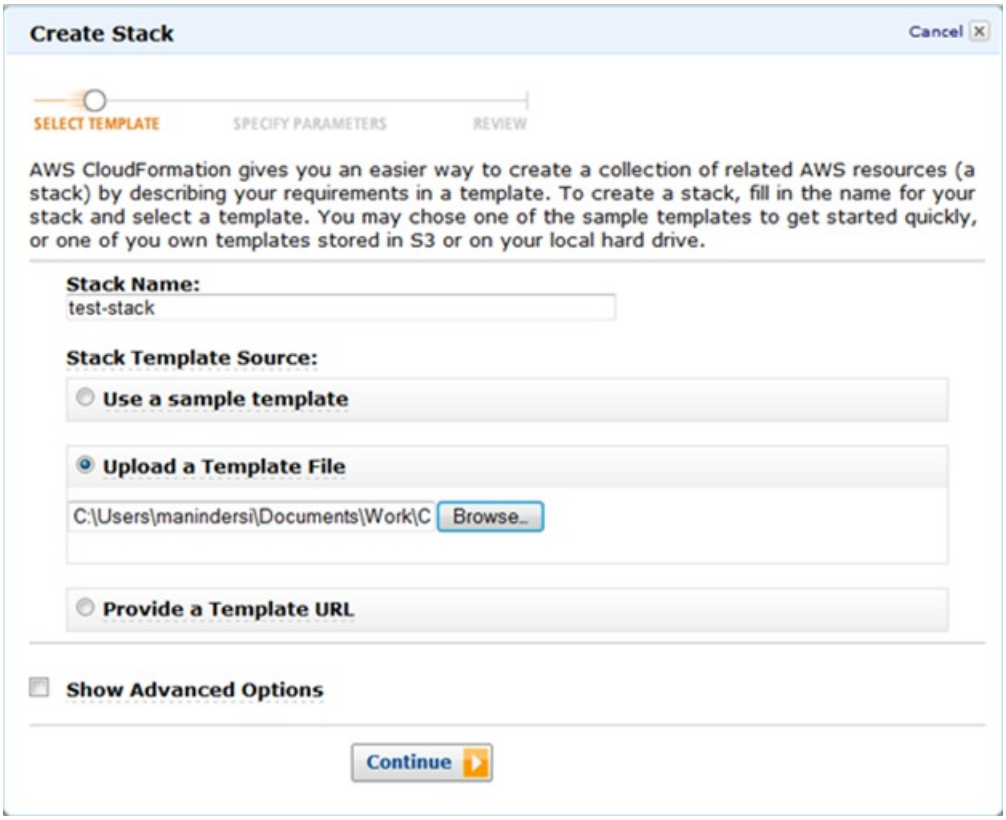
3. On the Amazon Web Services page, in the Deployment & Management section, click Cloud Formation.



4. On the CloudFormation Stacks page, select the Region in which you plan to deploy the NetScaler VPX instance, and then click Create New Stack.

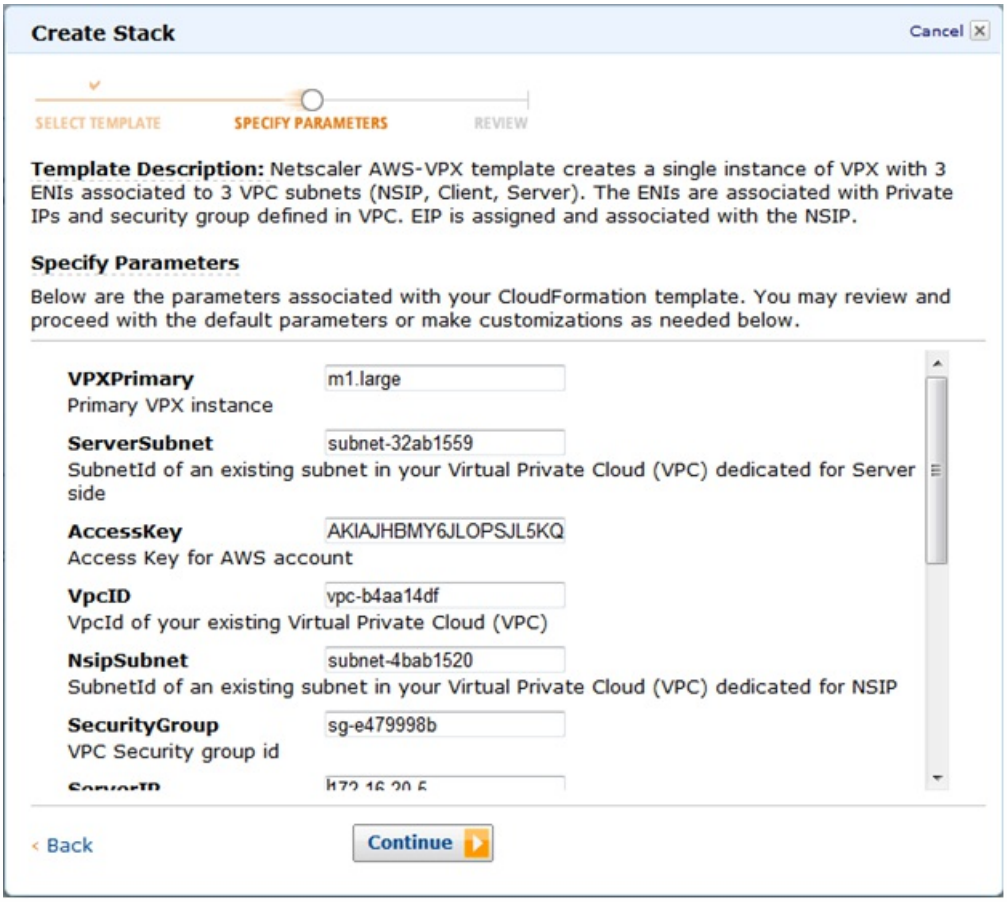


5. In the Create Stack dialog box, specify value for Stack Name, select the Upload a Template File option, and then click Browse. Select the template for HA NetScaler VPX from the local drive, and then click **Continue**.

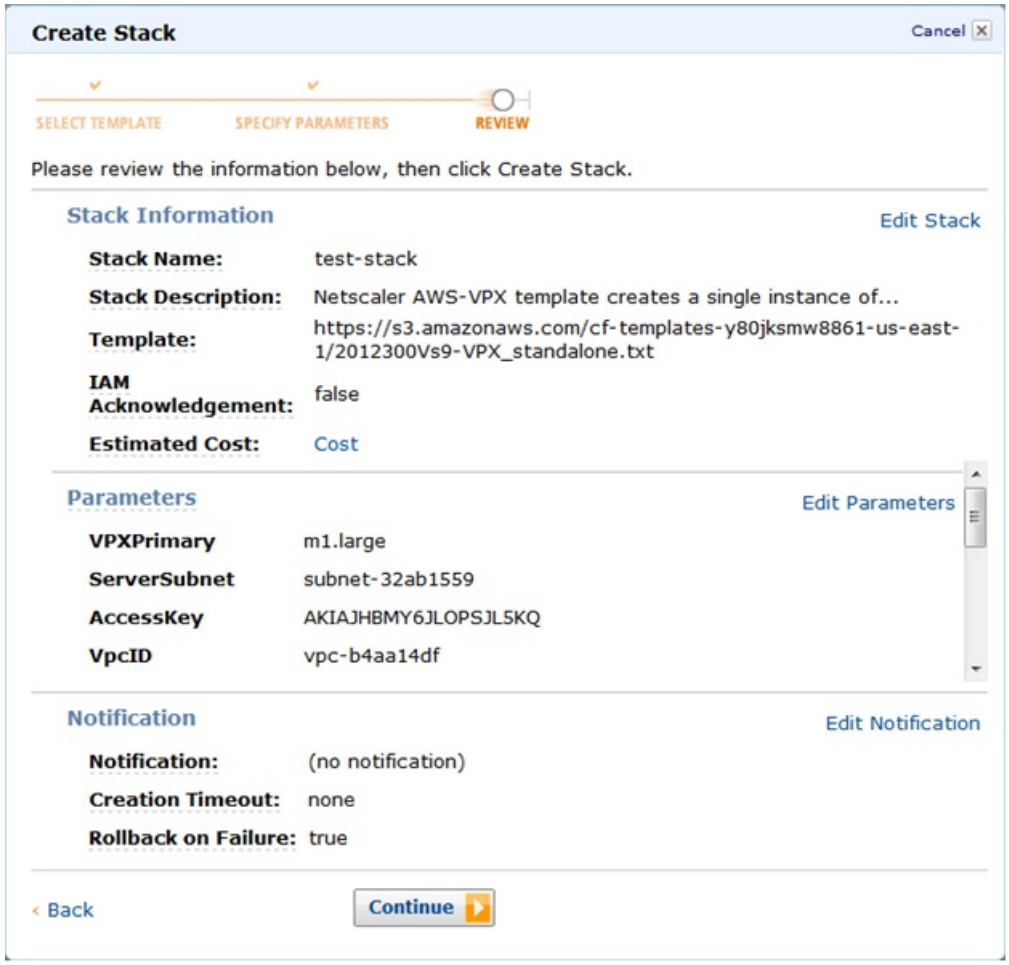


6. In the next pane, specify values for:
   - **VpcID**: An identifier to assign to the Virtual Private Cloud (VPC).
   - **NsipSubnet**: Subnet in which the NSIP is configured in VPC.
   - **ServerSubnet**: Subnet in which the server farm is configured in VPC.
   - **ClientSubnet**: SubnetId in which the client side is configured in VPC.
   - **SecurityGroup**: VPC Security group id.
   - **VPXPrimary**: Name of Primary VPX instance type.
   - **AccessKey**: Access Key for IAM user account.
   - **SecretKey**: Secret Key for IAM user account.
   - **TenancyType**: Instance tenancy type, can be default or dedicated.
   - **NsIP**: Private IP assigned to the NSIP ENI. The last octet of NSIP should be between 5 and 254.
   - **NsIPSec**: Private IP assigned to the NSIP ENI of Secondary. last octet has to be between 5 and 254.
   - **ServerIP**: Private IP assigned to the Server ENI. The last octet should be between 5 and 254.
   - **ClientIP**: Private IP assigned to the Client ENI. The last octet should be between 5 and 254.
   - **KeyName**: Name of an existing EC2 KeyPair to enable SSH access to the instances.

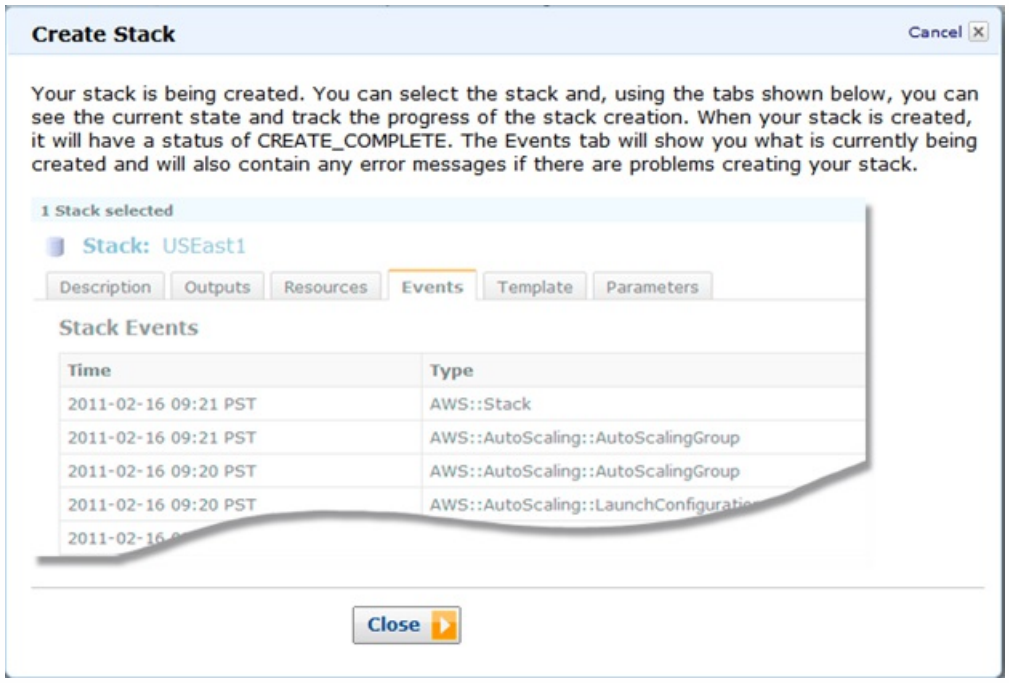
   Note: Make sure that the VPC, subnets, security groups, routes associations, gateway associations are already configured.



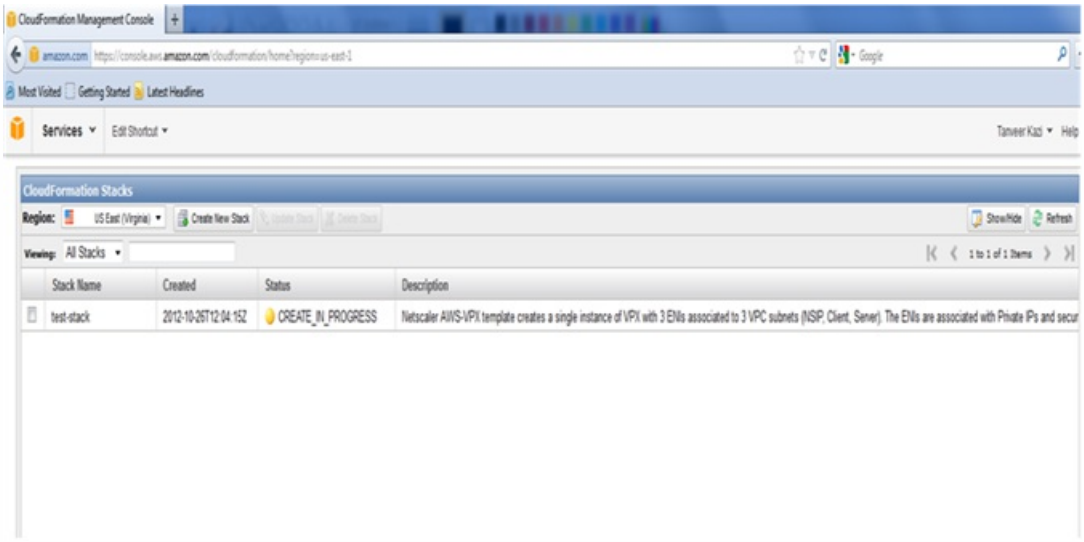
7. Click Continue.
8. Review the specified values in the Create Stack dialog box.



9. Click Continue to create a Stack.



10. Click Close to close the Create Stack dialog box.
11. The new stack that you created appears on the CloudFormation Stacks page.



Upgrading a NetScaler VPX instance on AWS

Apr 22, 2014

You can upgrade the EC2 instance type, throughput, software edition, and the system software of a NetScaler VPX running on AWS. For certain types of upgrades, Citrix recommends using the High Availability Configuration method to minimize downtime.

Note:

- NetScaler software release 10.1.e-124.1308.e or later for a NetScaler VPX AMI (including both utility license and customer license) does not support the M1 and M2 instance families.
- Because of changes in NetScaler instance support, downgrading from 10.1.e-124 or a later release to 10.1.123.x or an earlier release is not supported.
- Most of the upgrades do not require the launch of a new AMI, and the upgrade can be done on the current NetScaler AMI instance. If you do want to upgrade to a new NetScaler AMI instance, use the high availability configuration method.

Updated: 2014-04-22

If your NetScaler VPX instances are running release 10.1.e-124.1308.e or later, you can change the EC2 instance type from the AWS console as follows:

1. Stop the VPX instance.
2. Change the EC2 instance type from the AWS console.
3. Start the instance.

You can also use the above procedure to change the EC2 instance type for a release, earlier than 10.1.e-124.1308.e, unless you want to change the instance type to M3. In that case, you must first follow the standard NetScaler upgrade procedure, at Upgrading or Downgrading the System Software, to upgrade the NetScaler software to 10.1.e-124 or a later release, and then follow the above steps.

Updated: 2014-04-22

To upgrade the software edition (for example, to upgrade from standard to platinum edition) or throughput (for example, to upgrade from 200 mbps to 1000mbps), the method depends on the instance's license.

Using a customer license (Bring-Your-Own-License)

If you are using a customer license, you can purchase and download the new license from the Citrix Licensing portal (MyCitrix), and then install the license on the VPX instance. For more information about downloading and installing a license from the MyCitrix portal, see the VPX Licensing Guide.

Using a utility license (Utility license with hourly fee)

AWS does not support direct upgrades for fee-based instances. To upgrade the software edition or throughput of a fee based NetScaler VPX instance, launch a new AMI with the desired license and capacity and migrate the older instance configuration to the new instance. This can be achieved by using a NetScaler high availability configuration as described in "Upgrading to a New NetScaler AMI Instance by Using a NetScaler High Availability Configuration."

Updated: 2014-04-22

If you need to upgrade a NetScaler instance running 10.1.e-124.1308.e or a later release, follow the standard NetScaler upgrade procedure at Upgrading or Downgrading the System Software.

If you need to upgrade a NetScaler instance running a release older than 10.1.e-124.1308.e to 10.1.e-124.1308.e or a later release, first upgrade the system software, and then change the instance type to M3 as follows:

1. Stop the VPX instance.
2. Change the EC2 instance type from the AWS console.
3. Start the instance.

Updated: 2014-04-22

To use the high availability method of upgrading to a new NetScaler AMI instance, perform the following tasks:

- Create a new instance with the desired EC2 instance type, software edition, throughput, or software release from the AWS marketplace.
- Configure high availability between the old instance (to be upgraded) and the new instance. After high availability is configured between the old and the new instance, configuration from the old instance is synchronized to the new instance.
- Force an HA failover from the old instance to the new instance. As a result, the new instance becomes primary and starts receiving traffic.
- Stop, and reconfigure or remove the old instance from AWS.

Prerequisites and Points to Consider

- Make sure you understand how high availability works between two NetScaler VPX instances on AWS. For more information about high availability configuration between two NetScaler VPX instances on AWS, see High Availability.
- You must create the new instance in the same availability zone as the old instance, having the exact same security group and subnet.
- High availability setup requires access and secret keys associated with the user's AWS Identity and Access Management (IAM) account for both instances. If the correct key information is not used when creating VPX instances, the HA setup fails. For more information about creating an IAM account for a VPX instance, see Creating an IAM Account.
- You must use the EC2 console to create the new instance. You cannot use the AWS 1-click launch, because it does not accept the access and secret keys as the input.
- The new instance should have only one ENI interface.

To upgrade a NetScaler VPX Instance by using a high availability configuration

1. Configure high availability between the old and the new instance. To configure high availability between two NetScaler VPX instances, at the NetScaler command prompt of each intance, type:
   - add ha node <nodeID> <IPaddress of the node to be added>
   - save config

   Example

   At the NetScaler command prompt of the old instance, type:
   > add ha node 30 192.0.2.30
   Done

At the NetScaler command prompt of the new instance, type:

```
> add ha node 10 192.0.2.10
Done
```

Note the following:

- In the HA setup, the old instance is the primary node and the new instance is the secondary node.
- The NSIP IP address is not copied from the old instance to the new instance. Therefore, after the upgrade, your new instance has a different management IP address from the previous one.
- The nsroot account password of the new instance is set to that of the old instance after HA synchronization.

For more information about high availability configuration between two NetScaler VPX instances on AWS, see High Availability.

2. Force an HA failover. To force a failover in a high availability configuration, at the NetScaler command prompt of either of the instances, type:
   - force HA failover

   As the result of forcing a failover, the ENIs of the old instance are migrated to the new instance and traffic flows through the new instance (the new primary node). The old instance (the new secondary node) restarts.

   If the following warning message appears, type N to abort the operation:

   ```
   WARNING]:Force Failover may cause configuration loss, peer health not optimum. Reason(s):
   HA version mismatch
   HA heartbeats not seen on some interfaces
   Please confirm whether you want force-failover (Y/N)?
   ```

   The warning message appears because the system software of the two VPX instances is not HA compatible. As a result, the configuration of the old instance cannot be automatically synced to the new instance during a forced failover.

   Following is the workaround for this issue:

   1. At the NetScaler shell prompt of the old instance, type the following command to create a backup of the configuration file (ns.conf):
      - copy /nsconfig/ns.conf to /nsconfig/ns.conf.bkp
   2. Remove the following line from the backup configuration file (ns.conf.bkp):
      - set ns config -IPAddress <IP> -netmask <MASK>
        For example, set ns config -IPAddress 192.0.2.10 -netmask 255.255.255.0
   3. Copy the old instance's backup configuration file (ns.conf.bkp) to the /nsconfig directory of the the new instance.
   4. At the NetScaler shell prompt of the new instance, type the following command to load the old instance's configuration file (ns.conf.bkp) on the new instance:
      - batch -f /nsconfig/ns.conf.bkp
   5. Save the configuration on the new instance.
      - Save conifg
   6. At the NetScaler command prompt of either of the nodes, type the following command to force a failover, and then type Y for the warning message to confirm the force failover operation:
      - force ha failover

      Example

      ```
      > force ha failover

      WARNING]:Force Failover may cause configuration loss, peer health not optimum.
      ```

```
Reason(s):
HA version mismatch
HA heartbeats not seen on some interfaces
Please confirm whether you want force-failover (Y/N)? Y
```

3. Remove the HA configuration, so that the two instances are no longer in an HA configuration. First remove the HA configuration from the secondary node and then remove the HA configuration from the primary node.
   To remove an HA configuration between two NetScaler VPX instances, at the command prompt of each instance, type:

   - remove ha node <nodeID>
   - save config

   For more information about high availability configuration between two NetScaler instances on AWS, see High Availability.

   Example

   At the NetScaler command prompt of the old instance (new secondary node), type:

```
> remove ha node 30
  Done
> save config
  Done
```

   At the NetScaler command prompt of the new instance (new primary node), type:

```
> remove ha node 10
  Done
> save config
  Done
```

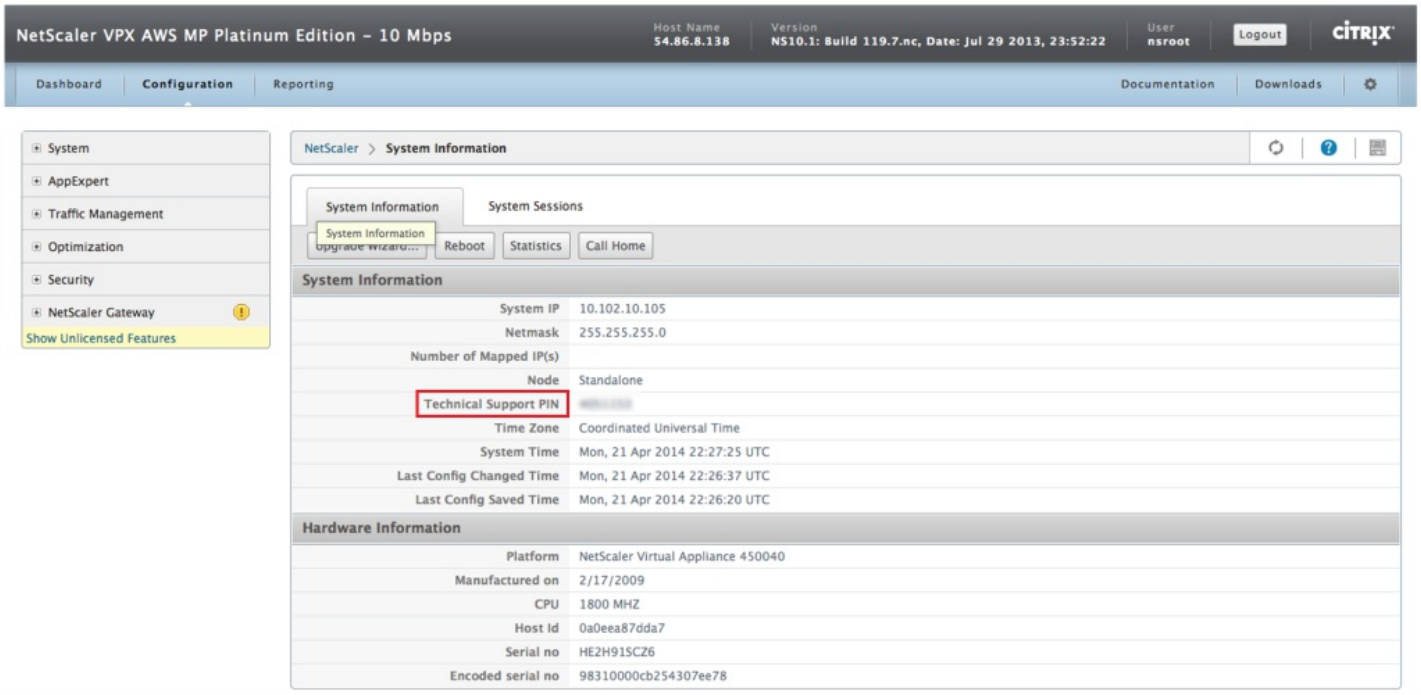
Troubleshooting the NetScaler VPX on AWS

Apr 28, 2014

Amazon does not provide console access to a NetScaler VPX virtual instance. To troubleshoot, you have to use the AWS GUI to view the activity log. You can debug only if the network is connected. To view an instance's system log, right-click the instance and select system log.

Citrix provides support for fee based NetScaler VPX instances (utility license with hourly fee) on AWS. To file a support case, find your AWS account number and support PIN code, and call Citrix support. You will also be asked for your name and email address. To find the support PIN, log on to the NetScaler configuration utility and navigate to the System page.

Here is an example of a system page showing the support PIN.



Configuring the Basic System Settings

Aug 23, 2013

After installing a Citrix NetScaler virtual appliance, you need to access it to configure the basic settings. Initially, you must access the NetScaler command line through the respective management application of the virtualization host (either Citrix XenCenter for Citrix XenServer or VMware vSphere client for VMware ESX) to specify a NetScaler IP (NSIP) address, subnet mask, and default gateway. The NSIP is the management address at which you can then access the NetScaler command line, through an SSH client, or access the configuration utility. You can use either of these access methods, or the console, to continue with basic configuration.

To access the configuration utility, type the NSIP into the address field of any browser (for example, http://<NSIP_address>). You need Java RunTime Environment (JRE) version 1.6 or later.

Updated: 2013-08-23

Your first task after installing a NetScaler virtual appliance on a virtualization host is to use the NetScaler virtual appliance console in the XenCenter client or vSphere client to configure the following initial settings.

Note: If you have installed a virtual appliance on XenServer by using Command Center, you do not have to configure these settings. Command Center implicitly configures the settings during installation. For more information about provisioning virtual appliance from Command Center, see the "Command Center" documentation.

NetScaler IP address (NSIP):

The IP address at which you access a NetScaler or a NetScaler virtual appliance for management purposes. A physical NetScaler or virtual appliance can have only one NSIP. You must specify this IP address when you configure the virtual appliance for the first time. You cannot remove an NSIP address.

Netmask:

The subnet mask associated with the NSIP address.

Default Gateway:

You must add a default gateway on the virtual appliance if you want access it through SSH or the configuration utility from an administrative workstation or laptop that is on a different network.

To configure the initial settings on the virtual appliance through the virtual appliance Console by using the management application

1. Connect to the XenServer or VMware ESX server on which the virtual appliance is installed by using XenCenter or vSphere client, respectively.
2. In the details pane, on the Console tab, log on to the virtual appliance by using the administrator credentials.
3. At the prompts, enter the NSIP address, subnet mask, and default gateway, and then save the configuration.

After you have set up an initial configuration through the NetScaler virtual appliance Console in the management application, you can use either the NetScaler command-line interface or the configuration utility to complete the configuration or to change the initial settings.

Updated: 2013-08-23

You can use the command line interface to set up the NSIP, Mapped IP (MIP), Subnet IP (SNIP), and hostname. You can also configure advanced network settings and change the time zone.

To complete initial configuration by using the command line interface

1. Use either the SSH client or the NetScaler virtual appliance Console in XenCenter to access the command line interface.
2. Log on to the virtual appliance, using the administrator credentials.
3. At the command prompt, type config ns to run the configuration script.
4. To complete the initial configuration, follow the prompts.

You have now completed the basic configuration of the virtual appliance. To continue the configuration process, choose one of the following options:

Citrix NetScaler Load Balancing Switch

If you are configuring the virtual appliance as a standard NetScaler load balancing switch with other licensed features, see "Traffic Management."

Citrix NetScaler Application Firewall

If you are configuring the virtual appliance as a standalone application firewall, see "Application Firewall."

For more information about the various features supported on the NetScaler virtual appliance, see Features at a Glance.

Updated: 2013-08-23

To use the Setup Wizard to set up the NetScaler virtual appliance, you must access the configuration utility from your Web browser. You can use the Setup Wizard to configure the NSIP, MIP, SNIP, hostname, and default gateway. You can also configure settings for a Web application by using an application template. You can also configure the appliance as a load balancer for Citrix XenDesktop or Citrix XenApp.

For information about application templates, see "AppExpert."

For information about the load balancing feature of a virtual appliance, see "Traffic Management."

To configure initial settings by using the configuration utility

1. In the address field of a Web browser, type: http://<NSIP address>
2. In User Name and Password, type the administrator credentials.
3. In Deployment Type, select NetScaler ADC.
4. In Start in, select Configuration, and then click Login.
5. In the Setup Wizard, click Next and follow the instructions.

You have now completed the basic configuration of the virtual appliance. To continue the configuration process, choose one of the following options:

Citrix NetScaler Load Balancing Switch

If you are configuring the virtual appliance as a standard NetScaler load balancing switch with other licensed features, see "Traffic Management."

Citrix NetScaler Application Firewall

If you are configuring the virtual appliance as a standalone application firewall, see "Application Firewall."

For more information about the various features supported on the NetScaler virtual appliance, see Features at a Glance.

Understanding Common Network Topologies

Sep 04, 2013

As described in "Physical Deployment Modes," you can deploy the Citrix NetScaler appliance either inline between the clients and servers or in one-arm mode. Inline mode uses a two-arm topology, which is the most common type of deployment.

This document includes the following:

- Setting Up Common Two-Arm Topologies
- Setting Up Common One-Arm Topologies

Updated: 2013-06-24

In a two-arm topology, one network interface is connected to the client network and another network interface is connected to the server network, ensuring that all traffic flows through the appliance. This topology might require you to reconnect your hardware and also might result in a momentary downtime. The basic variations of two-arm topology are multiple subnets, typically with the appliance on a public subnet and the servers on a private subnet, and transparent mode, with both the appliance and the servers on the public network.

Setting Up a Simple Two-Arm Multiple Subnet Topology

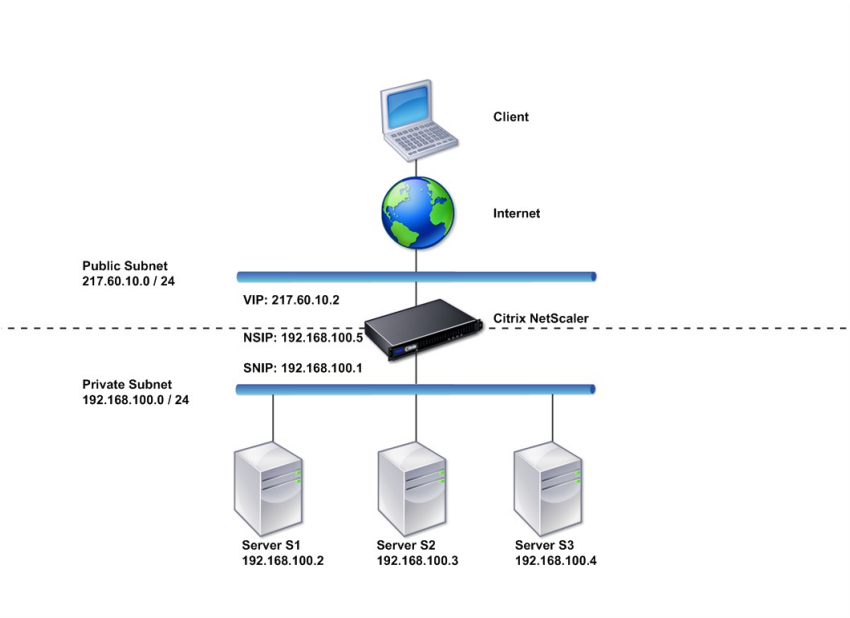
Updated: 2013-10-01

One of the most commonly used topologies has the NetScaler appliance inline between the clients and the servers, with a virtual server configured to handle the client requests. This configuration is used when the clients and servers reside on different subnets. In most cases, the clients and servers reside on public and private subnets, respectively.

For example, consider an appliance deployed in two-arm mode for managing servers S1, S2, and S3, with a virtual server of type HTTP configured on the appliance, and with HTTP services running on the servers. The servers are on a private subnet and a SNIP is configured on the appliance to communicate with the servers. The Use SNIP (USNIP) option must be enabled on the appliance so that it uses the SNIP instead of the MIP.

As shown in the following figure, the VIP is on public subnet 217.60.10.0, and the NSIP, the servers, and the SNIP are on private subnet 192.168.100.0/24.

Figure 1. Topology Diagram for Two-Arm Mode, Multiple Subnets



Task overview: To deploy a NetScaler appliance in two-arm mode with multiple subnets

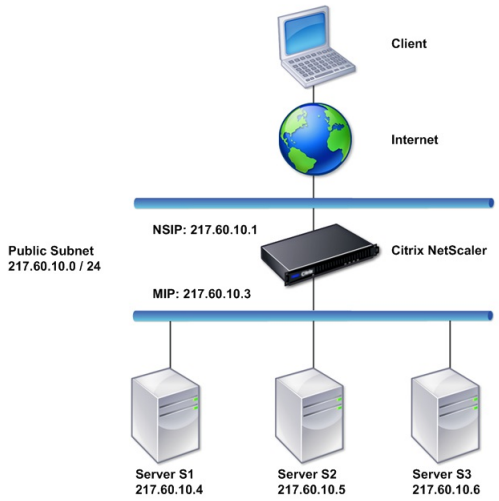
1. Configure the NSIP and default gateway, as described in "Configuring the NetScaler IP Address (NSIP)."
2. Configure the SNIP, as described in "Configuring Subnet IP Addresses."
3. Enable the USNIP option, as described in "To enable or disable USNIP mode."
4. Configure the virtual server and the services, as described in "Creating a Virtual Server" and "Configuring Services."
5. Connect one of the network interfaces to a private subnet and the other interface to a public subnet.

Setting Up a Simple Two-Arm Transparent Topology

Updated: 2013-09-16

Use transparent mode if the clients need to access the servers directly, with no intervening virtual server. The server IP addresses must be public because the clients need to be able to access them. In the example shown in the following figure, a NetScaler appliance is placed between the client and the server, so the traffic must pass through the appliance. You must enable L2 mode for bridging the packets. The NSIP and MIP are on the same public subnet, 217.60.10.0/24.

Figure 2. Topology Diagram for Two-Arm, Transparent Mode



Task overview: To deploy a NetScaler in two-arm, transparent mode

1. Configure the NSIP, MIP, and default gateway, as described in "Configuring a NetScaler by Using the Command Line Interface."
2. Enable L2 mode, as described in "Enabling and Disabling Layer 2 Mode."
3. Configure the default gateway of the managed servers as the MIP.
4. Connect the network interfaces to the appropriate ports on the switch.

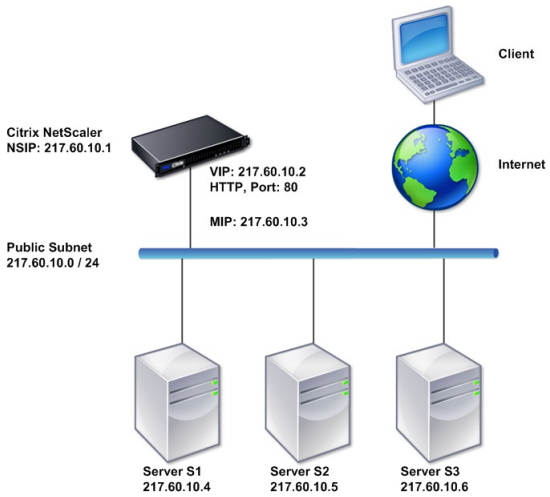
The two basic variations of one-arm topology are with a single subnet and with multiple subnets.

Setting Up a Simple One-Arm Single Subnet Topology

Updated: 2013-08-23

You can use a one-arm topology with a single subnet when the clients and servers reside on the same subnet. For example, consider a NetScaler deployed in one-arm mode for managing servers S1, S2, and S3. A virtual server of type HTTP is configured on a NetScaler, and HTTP services are running on the servers. As shown in the following figure, the NetScaler IP address (NSIP), the Mapped IP address (MIP), and the server IP addresses are on the same public subnet, 217.60.10.0/24.

Figure 3. Topology Diagram for One-Arm Mode, Single Subnet



Task overview: To deploy a NetScaler in one-arm mode with a single subnet

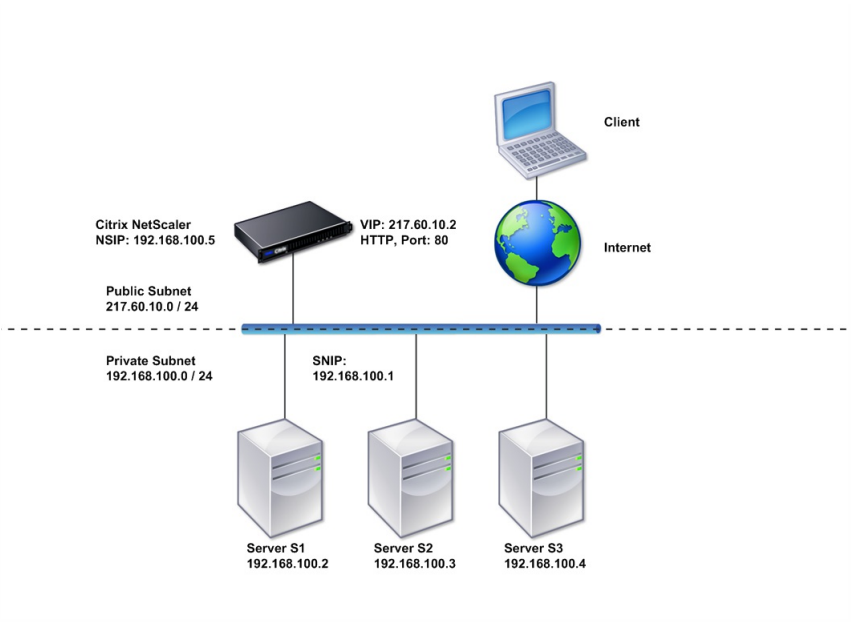
1. Configure the NSIP, MIP, and the default gateway, as described in "Configuring the NetScaler IP Address (NSIP)".
2. Configure the virtual server and the services, as described in "Creating a Virtual Server" and "Configuring Services".
3. Connect one of the network interfaces to the switch.

Setting Up a Simple One-Arm Multiple Subnet Topology

Updated: 2013-08-23

You can use a one-arm topology with multiple subnets when the clients and servers reside on the different subnets. For example, consider a NetScaler appliance deployed in one-arm mode for managing servers S1, S2, and S3, with the servers connected to switch SW1 on the network. A virtual server of type HTTP is configured on the appliance, and HTTP services are running on the servers. These three servers are on the private subnet, so a subnet IP address (SNIP) is configured to communicate with them. The Use Subnet IP address (USNIP) option must be enabled so that the appliance uses the SNIP instead of a MIP. As shown in the following figure, the virtual IP address (VIP) is on public subnet 217.60.10.0/24; the NSIP, SNIP, and the server IP addresses are on private subnet 192.168.100.0/24.

Figure 4. Topology Diagram for One-Arm Mode, Multiple Subnets



Task overview: To deploy a NetScaler appliance in one-arm mode with multiple subnets

1. Configure the NSIP and the default gateway, as described in "Configuring the NetScaler IP Address (NSIP)".
2. Configure the SNIP and enable the USNIP option, as described in "Configuring Subnet IP Addresses".
3. Configure the virtual server and the services, as described in "Creating a Virtual Server" and "Configuring Services".
4. Connect one of the network interfaces to the switch.

Configuring System Management Settings

Sep 04, 2013

Once your initial configuration is in place, you can configure settings to define the behavior of the Citrix NetScaler appliance and facilitate connection management. You have a number of options for handling HTTP requests and responses. Routing, bridging, and MAC based forwarding modes are available for handling packets not addressed to the NetScaler. You can define the characteristics of your network interfaces and can aggregate the interfaces. To prevent timing problems, you can synchronize the NetScaler clock with a Network Time Protocol (NTP) server. The NetScaler can operate in various DNS modes, including as an authoritative domain name server (ADNS). You can set up SNMP for system management and customize syslog logging of system events. Before deployment, verify that your configuration is complete and correct.

This document includes the following:

- Configuring System Settings
- Configuring Modes of Packet Forwarding
- Configuring Network Interfaces
- Configuring Clock Synchronization
- Configuring DNS
- Configuring SNMP
- Verifying the Configuration

Note: In addition to the tasks listed above, you can configure Syslog logging. For instructions, see "Audit Logging."

Configuring System Settings

Sep 04, 2013

Configuration of system settings includes basic tasks such as configuring HTTP ports to enable connection keep-alive and server offload, setting the maximum number of connections for each server, and setting the maximum number of requests per connection. You can enable client IP address insertion for situations in which a proxy IP address is not suitable, and you can change the HTTP cookie version.

You can also configure a NetScaler appliance to open FTP connections on a controlled range of ports instead of ephemeral ports for data connections. This improves security, because opening all ports on the firewall is insecure. You can set the range anywhere from 1,024 to 64,000.

Before deployment, go through the verification checklists to verify your configuration. To configure HTTP parameters and the FTP port range, use the NetScaler configuration utility.

You can modify the types of HTTP parameters described in the following table.

Table 1. HTTP Parameters

Parameter Type	Specifies
HTTP Port Information	The web server HTTP ports used by your managed servers. If you specify the ports, the appliance performs request switching for any client request that has a destination port matching a specified port. Note: If an incoming client request is not destined for a service or a virtual server that is specifically configured on the appliance, the destination port in the request must match one of the globally configured HTTP ports. This allows the appliance to perform connection keep-alive and server off-load.
Limits	The maximum number of connections to each managed server, and the maximum number of requests sent over each connection. For example, if you set Max Connections to 500, and the appliance is managing three servers, it can open a maximum of 500 connections to each of the three servers. By default, the appliance can create an unlimited number of connections to any of the servers it manages. To specify an unlimited number of requests per connection, set Max Requests to 0. Note: If you are using the Apache HTTP server, you must set Max Connections equal to the value of the MaxClients parameter in the Apache httpd.conf file. Setting this parameter is optional for other web servers.
Client IP Insertion	Enable/disable insertion of the client's IP address into the HTTP request header. You can specify a name for the header field in the adjacent text box. When a web server managed by an appliance receives a mapped IP address or a subnet IP address, the server identifies it as the client's IP address. Some applications need the client's IP address for logging purposes or to dynamically determine the content to be served by the web server. You can enable insertion of the actual client IP address into the HTTP header request sent from the client to one, some, or all servers managed by the appliance. You can then access the

Parameter Type	Specifies
	inserted address through a minor modification to the server (using an Apache module, ISAPI interface, or NSAPI interface).
Cookie Version	The HTTP cookie version to use when COOKIEINSERT persistence is configured on a virtual server. The default, version 0, is the most common type on the Internet. Alternatively, you can specify version 1.
Requests/Responses	Options for handling certain types of requests, and enable/disable logging of HTTP error responses.
Server Header Insertion	Insert a server header in NetScaler-generated HTTP responses.

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Settings, click Change HTTP parameters.
3. In the Configure HTTP parameters dialog box, specify values for some or all of the parameters that appear under the headings listed in the table above.
4. Click OK.

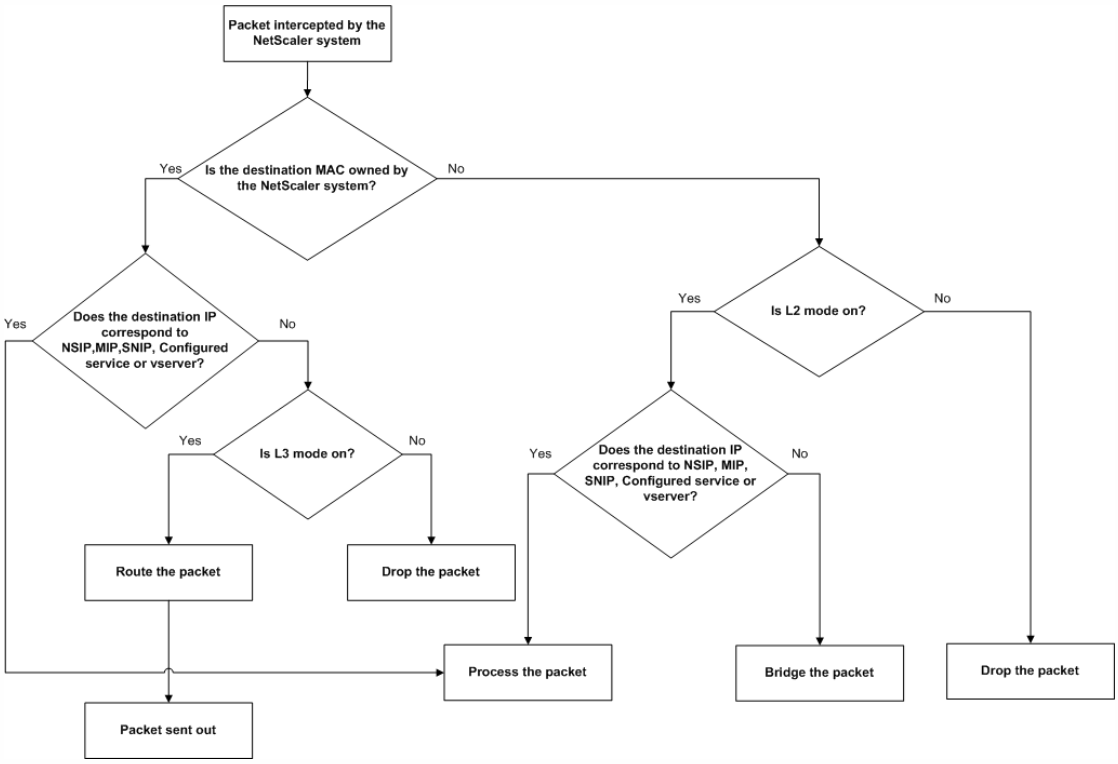
1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Settings, click Change global system settings.
3. Under FTP Port Range, in the Start Port and End Port text boxes, type the lowest and highest port numbers, respectively, for the range you want to specify (for example, 5000 and 6000).
4. Click OK.

Configuring Modes of Packet Forwarding

Jun 24, 2013

The NetScaler appliance can either route or bridge packets that are not destined for an IP address owned by the appliance (that is, the IP address is not the NSIP, a MIP, a SNIP, a configured service, or a configured virtual server). By default, L3 mode (routing) is enabled and L2 mode (bridging) is disabled, but you can change the configuration. The following flow chart shows how the appliance evaluates packets and either processes, routes, bridges, or drops them.

Figure 1. Interaction between Layer 2 and Layer 3 Modes



An appliance can use the following modes to forward the packets it receives:

- Layer 2 (L2) Mode
- Layer 3 (L3) Mode
- MAC-Based Forwarding Mode

Updated: 2013-09-13

Layer 2 mode controls the Layer 2 forwarding (bridging) function. You can use this mode to configure a NetScaler appliance to behave as a Layer 2 device and bridge the packets that are not destined for it. When this mode is enabled, packets are not forwarded to any of the MAC addresses, because the packets can arrive on any interface of the appliance and each interface has its own MAC address.

With Layer 2 mode disabled (which is the default), the appliance drops packets that are not destined for one of its MAC address. If another Layer 2 device is installed in parallel with the appliance, Layer 2 mode must be disabled to prevent

bridging (Layer 2) loops. You can use the configuration utility or the command line to enable Layer 2 mode.

Note: The appliance does not support spanning tree protocol. To avoid loops, if you enable L2 mode, do not connect two interfaces on the appliance to the same broadcast domain.

To enable or disable Layer 2 mode by using the command line interface

At the command prompt, type the following commands to enable/disable Layer 2 mode and verify that it has been enabled/disabled:

- enable ns mode <Mode>
- disable ns mode <Mode>
- show ns mode

Examples

```
> enable ns mode l2
 Done
> show ns mode

    Mode              Acronym       Status
    -------           -------       ------
 1)  Fast Ramp          FR           ON
 2)  Layer 2 mode        L2           ON
 .
 .
 .
 Done
>

> disable ns mode l2
 Done
> show ns mode

    Mode              Acronym       Status
    -------           -------       ------
 1)  Fast Ramp          FR           ON
 2)  Layer 2 mode        L2           OFF
 .
 .
 .
 Done
>
```

To enable or disable Layer 2 mode by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Configure modes.
3. In the Configure Modes dialog box, to enable Layer 2 mode, select the Layer 2 Mode check box. To disable Layer 2 mode, clear the check box.

4. Click OK. The Enable/Disable Mode(s)? message appears in the details pane.
5. Click Yes.

Updated: 2013-09-13

Layer 3 mode controls the Layer 3 forwarding function. You can use this mode to configure a NetScaler appliance to look at its routing table and forward packets that are not destined for it. With Layer 3 mode enabled (which is the default), the appliance performs route table lookups and forwards all packets that are not destined for any appliance-owned IP address. If you disable Layer 3 mode, the appliance drops these packets.

To enable or disable Layer 3 mode by using the command line interface

At the command prompt, type the following commands to enable/disable Layer 3 mode and verify that it has been enabled/disabled:

- enable ns mode <Mode>
- disable ns mode <Mode>
- show ns mode

Examples

```
> enable ns mode l3
 Done
> show ns mode

   Mode                 Acronym       Status
   -------              -------       ------
1) Fast Ramp              FR            ON
2) Layer 2 mode           L2            OFF
.
.
.
9) Layer 3 mode (ip forwarding)  L3          ON
.
.
.
 Done
>

> disable ns mode l3
 Done
> show ns mode

   Mode                 Acronym       Status
   -------              -------       ------
1) Fast Ramp              FR            ON
2) Layer 2 mode           L2            OFF
.
```


.
.
9) Layer 3 mode (ip forwarding) L3 OFF
.
.
.
Done
>

To enable or disable Layer 3 mode by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Configure modes.
3. In the Configure Modes dialog box, to enable Layer 3 mode, select the Layer 3 Mode (IP Forwarding) check box. To disable Layer 3 mode, clear the check box.
4. Click OK. The Enable/Disable Mode(s)? message appears in the details pane.
5. Click Yes.

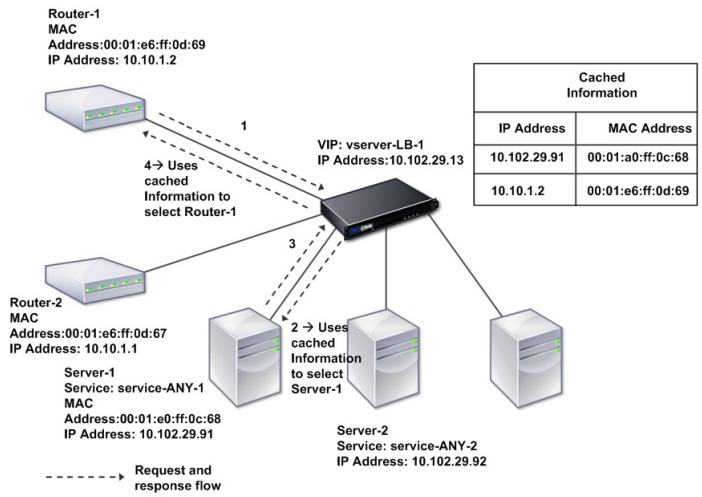
Updated: 2013-09-13

You can use MAC-based forwarding to process traffic more efficiently and avoid multiple-route or ARP lookups when forwarding packets, because the NetScaler appliance remembers the MAC address of the source. To avoid multiple lookups, the appliance caches the source MAC address of every connection for which it performs an ARP lookup, and it returns the data to the same MAC address.

MAC-based forwarding is useful when you use VPN devices because the appliance ensures that all traffic flowing through a particular VPN passes through the same VPN device.

The following figure shows the process of MAC-based forwarding.

Figure 2. MAC-Based Forwarding Process



When MAC-based forwarding is enabled, the appliance caches the MAC address of:

- The source (a transmitting device such as router, firewall, or VPN device) of the inbound connection.
- The server that responds to the requests.

When a server responds through an appliance, the appliance sets the destination MAC address of the response packet to the cached address, ensuring that the traffic flows in a symmetric manner, and then forwards the response to the client. The process bypasses the route table lookup and ARP lookup functions. However, when an appliance initiates a connection, it uses the route and ARP tables for the lookup function. To enable MAC-based forwarding, use the configuration utility or the command line.

Some deployments require the incoming and outgoing paths to flow through different routers. In these situations, MAC-based forwarding breaks the topology design. For a global server load balancing (GSLB) site that requires the incoming and outgoing paths to flow through different routers, you must disable MAC-based forwarding and use the appliance's default router as the outgoing router.

With MAC-based forwarding disabled and Layer 2 or Layer 3 connectivity enabled, a route table can specify separate routers for outgoing and incoming connections. To disable MAC-based forwarding, use the configuration utility or the command line.

To enable or disable MAC-based forwarding by using the command line interface

At the command prompt, type the following commands to enable/disable MAC-based forwarding mode and verify that it has been enabled/disabled:

- enable ns mode <Mode>
- disable ns mode <Mode>
- show ns mode

Example

```
> enable ns mode mbf
 Done
> show ns mode

   Mode                Acronym        Status
   -------             -------        ------
1) Fast Ramp             FR             ON
2) Layer 2 mode          L2             OFF
.
.
.
6) MAC-based forwarding      MBF            ON
.
.
.
Done
>

> disable ns mode mbf
```



```
Done
> show ns mode

   Mode                Acronym        Status
   -------             -------        ------
1) Fast Ramp             FR             ON
2) Layer 2 mode           L2             OFF
.
.
.
6) MAC-based forwarding      MBF            OFF
.
.
.
Done
>
```

To enable or disable MAC-based forwarding by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features group, click Configure modes.
3. In the Configure Modes dialog box, to enable MAC-based forwarding mode, select the MAC Based Forwarding check box. To disable MAC-based forwarding mode, clear the check box.
4. Click OK. The Enable/Disable Mode(s)? message appears in the details pane.
5. Click Yes.

Configuring Network Interfaces

Aug 23, 2013

NetScaler interfaces are numbered in slot/port notation. In addition to modifying the characteristics of individual interfaces, you can configure virtual LANs to restrict traffic to specific groups of hosts. You can also aggregate links into high-speed channels.

The NetScaler supports (Layer 2) port and IEEE802.1Q tagged virtual LANs (VLANs). VLAN configurations are useful when you need to restrict traffic to certain groups of stations. You can configure a network interface to belong to multiple VLANs by using IEEE 802.1q tagging.

You can bind your configured VLANs to IP subnets. The NetScaler (if it is configured as the default router for the hosts on the subnets) then performs IP forwarding between these VLANs. A NetScaler supports the following types of VLANs.

Default VLAN

By default, the network interfaces on a NetScaler are included in a single, port-based VLAN as untagged network interfaces. This default VLAN has a VID of 1 and exists permanently. It cannot be deleted, and its VID cannot be changed.

Port-Based VLANs

A set of network interfaces that share a common, exclusive, Layer 2 broadcast domain define the membership of a port-based VLAN. You can configure multiple port-based VLANs. When you add an interface to a new VLAN as an untagged member, it is automatically removed from the default VLAN.

Tagged VLAN

A network interface can be a tagged or untagged member of a VLAN. Each network interface is an untagged member of only one VLAN (its native VLAN). The untagged network interface forwards the frames for the native VLAN as untagged frames. A tagged network interface can be a part of more than one VLAN. When you configure tagging, be sure that both ends of the link have matching VLAN settings. You can use the configuration utility to define a tagged VLAN (nsvlan) that can have any ports bound as tagged members of the VLAN. Configuring this VLAN requires a reboot of the NetScaler and therefore must be done during initial network configuration.

Link aggregation combines incoming data from multiple ports into a single high speed link. Configuring the link aggregate channel increases the capacity and availability of the communication channel between a NetScaler and other connected devices. An aggregated link is also referred to as a channel.

When a network interface is bound to a channel, the channel parameters have precedence over the network interface parameters. A network interface can be bound to only one channel. Binding a network interface to a link aggregate channel changes the VLAN configuration. That is, binding network interfaces to a channel removes them from the VLANs that they originally belonged to and adds them to the default VLAN. However, you can bind the channel back to the old VLAN, or to a new one. For example, if you have bound network interfaces 1/2 and 1/3 to a VLAN with ID 2, and then you bind them to link aggregate channel LA/1, the network interfaces are moved to the default VLAN, but you can bind them to VLAN 2.

Note: You can also use Link Aggregation Control Protocol (LACP) to configure link aggregation. For more information, see ""Configuring Link Aggregation by Using the Link Aggregation Control Protocol."

Configuring Clock Synchronization

Aug 23, 2013

You can configure your NetScaler appliance to synchronize its local clock with a Network Time Protocol (NTP) server. This ensures that its clock has the same date and time settings as the other servers on your network. NTP uses User Datagram Protocol (UDP) port 123 as its transport layer. You have to add NTP servers in the NTP configuration file so that the appliance periodically gets updates from these servers.

If you do not have a local NTP server, you can find a list of public, open access, NTP servers at the official NTP site at http://www.ntp.org.

1. Log on to the command line and enter the shell command.
2. At the shell prompt, copy the ntp.conf file from the /etc directory to the /nsconfig directory. If the file already exists in the /nsconfig directory, make sure that you remove the following entries from the ntp.conf file:
   restrict localhost

   restrict 127.0.0.2

   These entries are required only if you want to run the device as a time server. However, this feature is not supported on the NetScaler.
3. Edit /nsconfig/ntp.conf by typing the IP address for the desired NTP server under the file's server and restrict entries.
4. Create a file named rc.netscaler in the /nsconfig directory, if the file does not already exist in the directory.
5. Edit /nsconfig/rc.netscaler by adding the following entry: /usr/sbin/ntpd -c /nsconfig/ntp.conf -l /var/log/ntpd.log &
   This entry starts the ntpd service, checks the ntp.conf file, and logs messages in the /var/log directory.

   Note: If the time difference between the NetScaler and the time server is more than 1000 sec, the ntpd service terminates with a message to the NetScaler log. To avoid this, you need to start ntpd with the -g option, which forcibly syncs the time. Add the following entry in /nsconfig/rc.netscaler:
   /usr/sbin/ntpd -g -c /nsconfig/ntp.conf -l /var/log/ntpd.log &

   If you do not want to forcibly sync the time when there is a large difference, you can set the date manually and then start ntpd again. You can check the time difference between the appliance and the time server by running the following command in the shell:

   ntpdate -q <IP address or domain name of the NTP server>
6. Reboot the appliance to enable clock synchronization.
   Note: If you want to start time synchronization before you restart the appliance, enter the following command (which you added to the rc.netscaler file in step 5) at the shell prompt:
   /usr/sbin/ntpd -c /nsconfig/ntp.conf -l /var/log/ ntpd.log &

Configuring DNS

Sep 04, 2013

You can configure a NetScaler appliance to function as an Authoritative Domain Name Server (ADNS), DNS proxy server, End Resolver, or Forwarder. You can add DNS resource records such as SRV Records, AAAA Records, A Records, MX Records, NS Records, CNAME Records, PTR Records, and SOA Records. Also, the appliance can balance the load on external DNS servers.

A common practice is to configure an appliance as a forwarder. For this configuration, you need to add external name servers. After you have added the external servers, you should verify that your configuration is correct.

You can add, remove, enable, and disable external name servers. You can create a name server by specifying its IP address, or you can configure an existing virtual server as the name server.

When adding name servers, you can specify IP addresses or virtual IP addresses (VIPs). If you use IP addresses, the appliance load balances requests to the configured name servers in a round robin manner. If you use VIPs, you can specify any load balancing method.

At the command prompt, type the following commands to add a name server and verify the configuration:

- add dns nameServer <IP>
- show dns nameServer <IP>

Example

```
> add dns nameServer 10.102.29.10
 Done
> show dns nameServer 10.102.29.10
1)      10.102.29.10  -  State: DOWN
 Done
>
```

1. Navigate to Traffic Management > DNS > Name Servers.
2. In the details pane, click Add.
3. In the Create Name Server dialog box, select IP Address.
4. In the IP Address text box, type the IP address of the name server (for example, 10.102.29.10). If you are adding an external name server, clear the Local check box.
5. Click Create, and then click Close.
6. Verify that the name server you added appears in the Name Servers pane.

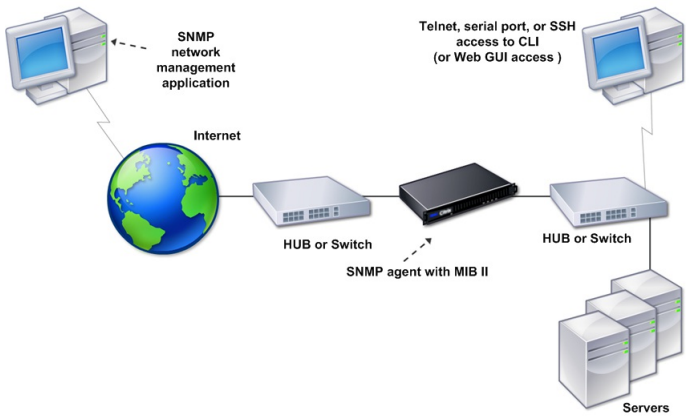
Configuring SNMP

May 14, 2012

The Simple Network Management Protocol (SNMP) network management application, running on an external computer, queries the SNMP agent on the NetScaler. The agent searches the management information base (MIB) for data requested by the network management application and sends the data to the application.

SNMP monitoring uses traps messages and alarms. SNMP traps messages are asynchronous events that the agent generates to signal abnormal conditions, which are indicated by alarms. For example, if you want to be informed when CPU utilization is above 90 percent, you can set up an alarm for that condition. The following figure shows a network with a NetScaler that has SNMP enabled and configured.

Figure 1. SNMP on the NetScaler



The SNMP agent on a NetScaler supports SNMP version 1 (SNMPv1), SNMP version 2 (SNMPv2), and SNMP version 3 (SNMPv3). Because it operates in bilingual mode, the agent can handle SNMPv2 queries, such as Get-Bulk, and SNMPv1 queries. The SNMP agent also sends traps compliant with SNMPv2 and supports SNMPv2 data types, such as counter64. SNMPv1 managers (programs on other servers that request SNMP information from the NetScaler) use the NS-MIB-smiv1.mib file when processing SNMP queries. SNMPv2 managers use the NS-MIB-smiv2.mib file.

The NetScaler supports the following enterprise-specific MIBs:

A subset of standard MIB-2 groups

Provides MIB-2 groups SYSTEM, IF, ICMP, UDP, and SNMP.

A system enterprise MIB

Provides system-specific configuration and statistics.

To configure SNMP, you specify which managers can query the SNMP agent, add SNMP trap listeners that will receive the SNMP trap messages, and configure SNMP Alarms.

Updated: 2013-06-05

You can configure a workstation running a management application that complies with SNMP version 1, 2, or 3 to access an appliance. Such a workstation is called an SNMP manager. If you do not specify an SNMP manager on the appliance, the appliance accepts and responds to SNMP queries from all IP addresses on the network. If you configure one or more SNMP managers, the appliance accepts and responds to SNMP queries from only those specific IP addresses. When specifying the IP address of an SNMP manager, you can use the netmask parameter to grant access from entire subnets. You can add a maximum of 100 SNMP managers or networks.

To add an SNMP manager by using the command line interface

At the command prompt, type the following commands to add an SNMP manager and verify the configuration:

- add snmp manager <IPAddress> ... [-netmask <netmask>]
- show snmp manager <IPAddress>

Example

```
> add snmp manager 10.102.29.5 -netmask 255.255.255.255
 Done
> show snmp manager 10.102.29.5
1)     10.102.29.5        255.255.255.255
 Done
>
```

To add an SNMP manager by using the configuration utility

1. In the navigation pane, expand System, expand SNMP, and then click Managers.
2. In the details pane, click Add.
3. In the Add SNMP Manager dialog box, in the IP Address text box, type the IP address of the workstation running the management application (for example, 10.102.29.5).
4. Click Create, and then click Close.
5. Verify that the SNMP manager you added appears in the Details section at the bottom of the pane.

Updated: 2013-09-13

After configuring the alarms, you need to specify the trap listener to which the appliance will send the trap messages. Apart from specifying parameters like IP address and the destination port of the trap listener, you can specify the type of trap (either generic or specific) and the SNMP version.

You can configure a maximum of 20 trap listeners for receiving either generic or specific traps.

To add an SNMP trap listener by using the command line interface

At the command prompt, type the following command to add an SNMP trap and verify that it has been added:

- add snmp trap specific <IP>
- show snmp trap

Example

```
> add snmp trap specific 10.102.29.3
```

```
 Done
> show snmp trap
Type    DestinationIP   DestinationPort  Version  SourceIP     Min-Severity  Community
----    -------------   ---------------  -------  --------     ------------  ---------
generic  10.102.29.9    162          V2     NetScaler IP  N/A        public
generic  10.102.29.5    162          V2     NetScaler IP  N/A        public
generic  10.102.120.101  162          V2     NetScaler IP  N/A        public
.
.
.
specific 10.102.29.3    162          V2     NetScaler IP   -         public
 Done
>
```

To add an SNMP trap listener by using the configuration utility

1. In the navigation pane, expand System, expand SNMP, and then click Traps.
2. In the details pane, click Add.
3. In the Create SNMP Trap Destination dialog box, in the Destination IP Address text box, type the IP address (for example, 10.102.29.3).
4. Click Create and then click Close.
5. Verify that the SNMP trap you added appears in the Details section at the bottom of the pane.

Updated: 2013-09-13

You configure alarms so that the appliance generates a trap message when an event corresponding to one of the alarms occurs. Configuring an alarm consists of enabling the alarm and setting the severity level at which a trap is generated. There are five severity levels: Critical, Major, Minor, Warning, and Informational. A trap is sent only when the severity of the alarm matches the severity specified for the trap.

Some alarms are enabled by default. If you disable an SNMP alarm, the appliance will not generate trap messages when corresponding events occur. For example, if you disable the Login-Failure SNMP alarm, the appliance will not generate a trap message when a login failure occurs.

To enable or disable an alarm by using the command line interface

At the command prompt, type the following commands to enable or disable an alarm and verify that it has been enabled or disabled:

- set snmp alarm <trapName> [-state ENABLED | DISABLED]
- show snmp alarm <trapName>

Example

```
> set snmp alarm LOGIN-FAILURE -state ENABLED
 Done
> show snmp alarm LOGIN-FAILURE
  Alarm        Alarm Threshold  Normal Threshold  Time  State    Severity   Logging
```



```
  -----         ---------------  ----------------  ----  --------  ---------  --------
 1) LOGIN-FAILURE  N/A           N/A           N/A   ENABLED   -       ENABLED
 Done
>
```

To set the severity of the alarm by using the command line interface

At the command prompt, type the following commands to set the severity of the alarm and verify that the severity has been set correctly:

- set snmp alarm <trapName> [-severity <severity>]
- show snmp alarm <trapName>

Example

```
> set snmp alarm LOGIN-FAILURE -severity Major
 Done
> show snmp alarm LOGIN-FAILURE
  Alarm         Alarm Threshold  Normal Threshold  Time  State    Severity   Logging
  -----         ---------------  ----------------  ----  --------  ---------  --------
 1) LOGIN-FAILURE  N/A           N/A           N/A   ENABLED   Major     ENABLED
 Done
>
```

To configure alarms by using the configuration utility

1. In the navigation pane, expand System, expand SNMP, and then click Alarms.
2. In the details pane, select an alarm (for example, LOGIN-FAILURE), and then click Open.
3. In the Configure SNMP Alarm dialog box, to enable the alarm, select Enabled in the State drop-down list. To disable the alarm, select Disabled.
4. In the Severity drop-down list, select a severity option (for example, Major).
5. Click OK, and then click Close.
6. Verify that the parameters for the SNMP alarm you configured are correctly configured by viewing the Details section at the bottom of the pane.

Verifying the Configuration

Nov 08, 2013

After you finish configuring your system, complete the following checklists to verify your configuration.

Configuration Checklist

- The build running is:
- There are no incompatibility issues. (Incompatibility issues are documented in the build's release notes.)
- The port settings (speed, duplex, flow control, monitoring) are the same as the switch's port.
- Enough mapped IP addresses have been configured to support all server-side connections during peak times.
  - The number of configured mapped IP addresses is: _____
  - The expected number of simultaneous server connections is:
    [ ] 62,000 [ ] 124,000 [ ] Other_____

Topology Configuration Checklist

- The routes have been used to resolve servers on other subnets.

  The routes entered are:

  _____

- If the NetScaler is in a public-private topology, reverse NAT has been configured.
- The failover (high availability) settings configured on the NetScaler resolve in a one arm or two-arm configuration. All unused network interfaces have been disabled:

  _____

- If the NetScaler is placed behind an external load balancer, then the load balancing policy on the external load balancer is not "least connection."
  The load balancing policy configured on the external load balancer is:

  _____

- If the NetScaler is placed in front of a firewall, the session time-out on the firewall is set to a value greater than or equal to 300 seconds.
  Note: The TCP idle connection timeout on a NetScaler appliance is 360 seconds. If the timeout on the firewall is also set to 300 seconds or more, then the appliance can perform TCP connection multiplexing effectively because connections will not be closed earlier.
  The value configured for the session time-out is: _____

Server Configuration Checklist

- "Keep-alive" has been enabled on all the servers.
  The value configured for the keep-alive time-out is: _____

- The default gateway has been set to the correct value. (The default gateway should either be a NetScaler or upstream router.) The default gateway is:

  _____

- The server port settings (speed, duplex, flow control, monitoring) are the same as the switch port settings.

  _____

- If the Microsoft® Internet Information Server is used, buffering is enabled on the server.
- If an Apache Server is used, the MaxConn (maximum number of connections) parameter is configured on the server and on the NetScaler.
  The MaxConn (maximum number of connections) value that has been set is:

  _____

- If a Netscape® Enterprise Server™ is used, the maximum requests per connection parameter is set on the NetScaler. The maximum requests per connection value that has been set is:

  _____

Software Features Configuration Checklist

- Does the Layer 2 mode feature need to be disabled? (Disable if another Layer 2 device is working in parallel with a NetScaler.)
  Reason for enabling or disabling:

  _____

- Does the MAC-based forwarding feature need to be disabled? (If the MAC address used by return traffic is different, it should be disabled.)
  Reason for enabling or disabling:

  _____

- Does host-based reuse need to be disabled? (Is there virtual hosting on the servers?)
  Reason for enabling or disabling:

  _____

- Do the default settings of the surge protection feature need to be changed?
  Reason for changing or not changing:

  _____

Access Checklist

- The system IPs can be pinged from the client-side network.
- The system IPs can be pinged from the server-side network.
- The managed server(s) can be pinged through the NetScaler.
- Internet hosts can be pinged from the managed servers.
- The managed server(s) can be accessed through the browser.
- The Internet can be accessed from managed server(s) using the browser.
- The system can be accessed using SSH.
- Admin access to all managed server(s) is working.

Note: When you are using the ping utility, ensure that the pinged server has ICMP ECHO enabled, or your ping will not succeed.

Firewall Checklist

The following firewall requirements have been met:

- UDP 161 (SNMP)
- UDP 162 (SNMP trap)
- TCP/UDP 3010 (GUI)
- HTTP 80 (GUI)
- TCP 22 (SSH)

Load Balancing Traffic on a NetScaler Appliance

Jun 24, 2013

The load balancing feature distributes client requests across multiple servers to optimize resource utilization. In a real-world scenario with a limited number of servers providing service to a large number of clients, a server can become overloaded and degrade the performance of the server farm. A Citrix NetScaler appliance uses load balancing criteria to prevent bottlenecks by forwarding each client request to the server best suited to handle the request when it arrives.

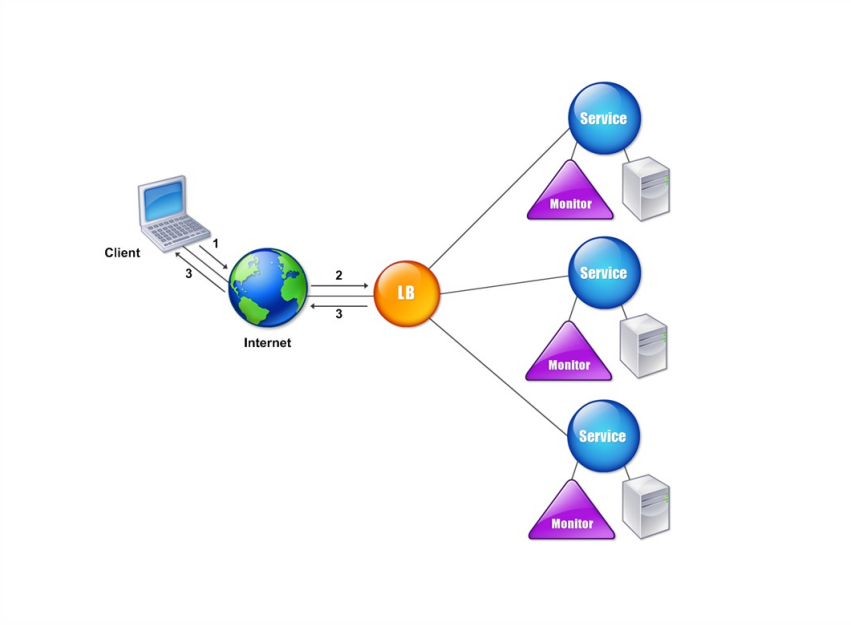
To configure load balancing, you define a virtual server to proxy multiple servers in a server farm and balance the load among them.

When a client initiates a connection to the server, a virtual server terminates the client connection and initiates a new connection with the selected server, or reuses an existing connection with the server, to perform load balancing. The load balancing feature provides traffic management from Layer 4 (TCP and UDP) through Layer 7 (FTP, HTTP, and HTTPS).

The NetScaler appliance uses a number of algorithms, called load balancing methods, to determine how to distribute the load among the servers. The default load balancing method is the Least Connections method.

A typical load balancing deployment consists of the entities described in the following figure.

Figure 1. Load Balancing Architecture



The entities function as follows:

- **Virtual server**. An entity that is represented by an IP address, a port, and a protocol. The virtual server IP address (VIP) is usually a public IP address. The client sends connection requests to this IP address. The virtual server represents a bank of servers.
- **Service**. A logical representation of a server or an application running on a server. Identifies the server's IP address, a port, and a protocol. The services are bound to the virtual servers.
- **Server object**. An entity that is represented by an IP address. The server object is created when you create a service. The IP address of the service is taken as the name of the server object. You can also create a server object and then create

services by using the server object.

- **Monitor**. An entity that tracks the health of the services. The appliance periodically probes the servers using the monitor bound to each service. If a server does not respond within a specified response timeout, and the specified number of probes fails, the service is marked DOWN. The appliance then performs load balancing among the remaining services.

Configuring Load Balancing

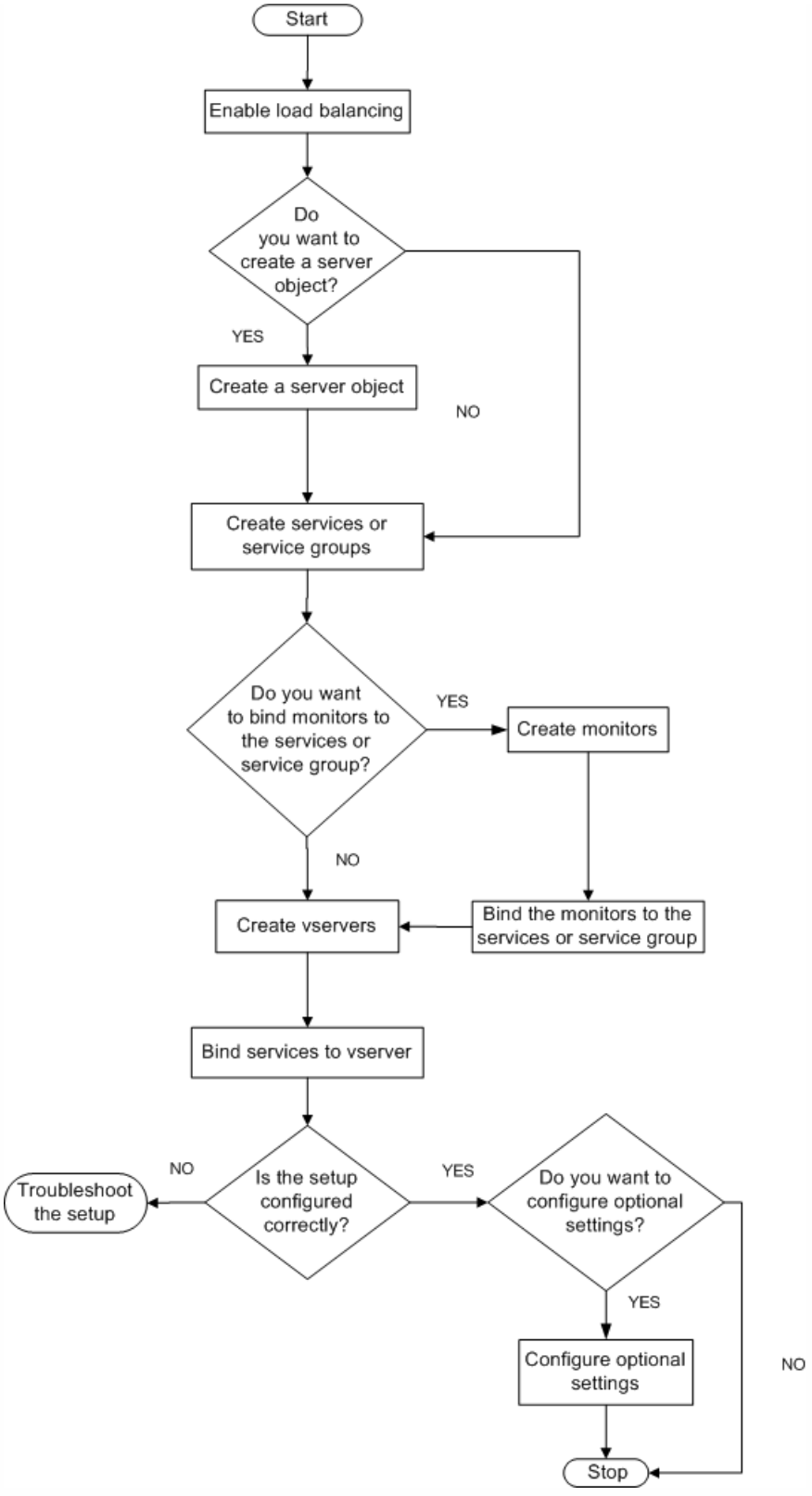
Jun 24, 2013

To configure load balancing, you must first create services. Then, you create virtual servers and bind the services to the virtual servers. By default, the NetScaler appliance binds a monitor to each service. After binding the services, verify your configuration by making sure that all of the settings are correct.

Note: After you deploy the configuration, you can display statistics that show how the entities in the configuration are performing. Use the statistical utility or the stat lb vserver <vserverName> command.
Optionally, you can assign weights to a service. The load balancing method then uses the assigned weight to select a service. For getting started, however, you can limit optional tasks to configuring some basic persistence settings, for sessions that must maintain a connection to a particular server, and some basic configuration-protection settings.

The following flow chart illustrates the sequence of the configuration tasks.

Figure 1. Sequence of Tasks to Configure Load Balancing



Updated: 2013-06-05

Before configuring load balancing, make sure that the load balancing feature is enabled.

To enable load balancing by using the command line interface

At the command prompt, type the following commands to enable load balancing and verify that it is enabled:

- enable feature lb
- show feature

Example

```
> enable feature lb
 Done
> show feature

   Feature             Acronym       Status
   -------             -------       ------
 1) Web Logging          WL            OFF
 2) Surge Protection     SP            OFF
 3) Load Balancing       LB            ON
 .
 .
 .
 9) SSL Offloading       SSL           ON
 .
 .
 .
Done
```

To enable load balancing by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Load Balancing check box, and then click OK.
4. In the Enable/Disable Feature(s)? message, click Yes.

Updated: 2013-06-24

When you have identified the services you want to load balance, you can implement your initial load balancing configuration by creating the service objects, creating a load balancing virtual server, and binding the service objects to the virtual server.

To implement the initial load balancing configuration by using the command line interface

At the command prompt, type the following commands to implement and verify the initial configuration:

- add service <name> <IPaddress> <serviceType> <port>
- add lb vserver <vServerName> <serviceType> [<IPaddress> <port>]
- bind lb vserver <name> <serviceName>

- show service bindings <serviceName>
  Example

```
> add service service-HTTP-1 10.102.29.5 HTTP 80
 Done
> add lb vserver vserver-LB-1 HTTP 10.102.29.60 80
 Done
> bind lb vserver vserver-LB-1 service-HTTP-1
 Done
> show service bindings service-HTTP-1
   service-HTTP-1 (10.102.29.5:80) - State : DOWN

      1)     vserver-LB-1 (10.102.29.60:80) - State : DOWN
 Done
```

To implement the initial load balancing configuration by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. In the details pane, under Getting Started, click Load Balancing wizard, and follow the instructions to create a basic load balancing setup.
3. Return to the navigation pane, expand Load Balancing, and then click Virtual Servers.
4. Select the virtual server that you configured and verify that the parameters displayed at the bottom of the page are correctly configured.
5. Click Open.
6. Verify that each service is bound to the virtual server by confirming that the Active check box is selected for each service on the Services tab.

Choosing and Configuring Persistence Settings

Sep 04, 2013

You must configure persistence on a virtual server if you want to maintain the states of connections on the servers represented by that virtual server (for example, connections used in e-commerce). The appliance then uses the configured load balancing method for the initial selection of a server, but forwards to that same server all subsequent requests from the same client.

If persistence is configured, it overrides the load balancing methods once the server has been selected. If the configured persistence applies to a service that is down, the appliance uses the load balancing methods to select a new service, and the new service becomes persistent for subsequent requests from the client. If the selected service is in an Out Of Service state, it continues to serve the outstanding requests but does not accept new requests or connections. After the shutdown period elapses, the existing connections are closed. The following table lists the types of persistence that you can configure.

Table 1. Limitations on Number of Simultaneous Persistent Connections

Persistence Type	Persistent Connections
Source IP, SSL Session ID, Rule, DESTIP, SRCIPDESTIP	250K
CookieInsert, URL passive, Custom Server ID	Memory limit. In case of CookieInsert, if time out is not 0, any number of connections is allowed until limited by memory.

If the configured persistence cannot be maintained because of a lack of resources on an appliance, the load balancing methods are used for server selection. Persistence is maintained for a configured period of time, depending on the persistence type. Some persistence types are specific to certain virtual servers. The following table shows the relationship.

Table 2. Persistence Types Available for Each Type of Virtual Server

Persistence TypeHeader 1	HTTP	HTTPS	TCP	UDP/IP	SSL_Bridge
Source IP	YES	YES	YES	YES	YES
CookieInsert	YES	YES	NO	NO	NO
SSL Session ID	NO	YES	NO	NO	YES
URL Passive	YES	YES	NO	NO	NO
Custom Server ID	YES	YES	NO	NO	NO
Rule	YES	YES	NO	NO	NO

Persistence Type	HTTP	HTTPS	TCP	UDP/IP	SSL_Bridge
SRCIPDESTIP	N/A	N/A	YES	YES	N/A
Header 1					
DESTIP	N/A	N/A	YES	YES	N/A

You can also specify persistence for a group of virtual servers. When you enable persistence on the group, the client requests are directed to the same selected server regardless of which virtual server in the group receives the client request. When the configured time for persistence elapses, any virtual server in the group can be selected for incoming client requests.

Two commonly used persistence types are persistence based on cookies and persistence based on server IDs in URLs.

Updated: 2013-08-23

When you enable persistence based on cookies, the NetScaler adds an HTTP cookie into the Set-Cookie header field of the HTTP response. The cookie contains information about the service to which the HTTP requests must be sent. The client stores the cookie and includes it in all subsequent requests, and the NetScaler uses it to select the service for those requests. You can use this type of persistence on virtual servers of type HTTP or HTTPS.

The NetScaler inserts the cookie <NSC_XXXX>= <ServiceIP> <ServicePort>

where:

- <NSC_XXXX> is the virtual server ID that is derived from the virtual server name.
- <ServiceIP> is the hexadecimal value of the IP address of the service.
- <ServicePort> is the hexadecimal value of the port of the service.

The NetScaler encrypts ServiceIP and ServicePort when it inserts a cookie, and decrypts them when it receives a cookie.

Note: If the client is not allowed to store the HTTP cookie, the subsequent requests do not have the HTTP cookie, and persistence is not honored.
By default, the NetScaler sends HTTP cookie version 0, in compliance with the Netscape specification. It can also send version 1, in compliance with RFC 2109.

You can configure a timeout value for persistence that is based on HTTP cookies. Note the following:

- If HTTP cookie version 0 is used, the NetScaler inserts the absolute Coordinated Universal Time (GMT) of the cookie's expiration (the expires attribute of the HTTP cookie), calculated as the sum of the current GMT time on a NetScaler, and the timeout value.
- If an HTTP cookie version 1 is used, the NetScaler inserts a relative expiration time (Max-Age attribute of the HTTP cookie). In this case, the client software calculates the actual expiration time.

Note: Most client software currently installed (Microsoft Internet Explorer and Netscape browsers) understand HTTP cookie version 0; however, some HTTP proxies understand HTTP cookie version 1.
If you set the timeout value to 0, the NetScaler does not specify the expiration time, regardless of the HTTP cookie version used. The expiration time then depends on the client software, and such cookies are not valid if that software is shut down. This persistence type does not consume any system resources. Therefore, it can accommodate an unlimited number of persistent clients.

An administrator can use the procedure in the following table to change the HTTP cookie version.

To change the HTTP cookie version by using the configuration utility

1. Navigate to System > Settings.
2. In the details pane, click Change HTTP Parameters.
3. In the Configure HTTP Parameters dialog box, under Cookie, select Version 0 or Version 1.

Note: For information about the parameters, see "Configuring Persistence Based on Cookies."

To configure persistence based on cookies by using the command line interface

At the command prompt, type the following commands to configure persistence based on cookies and verify the configuration:

- set lb vserver <name> -persistenceType COOKIEINSERT
- show lb vserver <name>

Example

```
> set lb vserver vserver-LB-1 -persistenceType COOKIEINSERT
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   .
   .
   .
   Persistence: COOKIEINSERT (version 0)   Persistence Timeout: 2 min
   .
   .
   .
 Done
>
```

To configure persistence based on cookies by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure persistence (for example, vserver-LB-1), and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Method and Persistence tab, in the Persistence list, select COOKIEINSERT.
4. In the Time-out (min) text box, type the time-out value (for example, 2).
5. Click OK.
6. Verify that the virtual server for which you configured persistence is correctly configured by selecting the virtual server and viewing the Details section at the bottom of the pane.

Updated: 2013-08-23

The NetScaler can maintain persistence based on the server IDs in the URLs. In a technique called URL passive persistence, the NetScaler extracts the server ID from the server response and embeds it in the URL query of the client request. The

server ID is an IP address and port specified as a hexadecimal number. The NetScaler extracts the server ID from subsequent client requests and uses it to select the server.

URL passive persistence requires configuring either a payload expression or a policy infrastructure expression specifying the location of the server ID in the client requests. For more information about expressions, see "Policy Configuration and Reference."

Note: If the server ID cannot be extracted from the client requests, server selection is based on the load balancing method.

Example: Payload Expression

The expression, URLQUERY contains sid= configures the system to extract the server ID from the URL query of a client request, after matching token sid=. Thus, a request with the URL http://www.citrix.com/index.asp?&sid;=c0a864100050 is directed to the server with the IP address10.102.29.10 and port 80.

The timeout value does not affect this type of persistence, which is maintained as long as the server ID can be extracted from the client requests. This persistence type does not consume any system resources, so it can accommodate an unlimited number of persistent clients.

Note: For information about the parameters, see "Load Balancing."

To configure persistence based on server IDs in URLs by using the command line interface

At the command prompt, type the following commands to configure persistence based on server IDs in URLs and verify the configuration:

- set lb vserver <name> -persistenceType URLPASSIVE
- show lb vserver <name>

Example

```
> set lb vserver vserver-LB-1 -persistenceType URLPASSIVE
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
    .
    .
    .
   Persistence: URLPASSIVE Persistence Timeout: 2 min
    .
    .
    .
 Done
>
```

To configure persistence based on server IDs in URLs by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure persistence (for example, vserver-LB-1), and then click Open.

3. In the Configure Virtual Server (Load Balancing) dialog box, on the Method and Persistence tab, in the Persistence list, select URLPASSIVE.
4. In the Time-out (min) text box, type the time-out value (for example, 2).
5. In the Rule text box, enter a valid expression. Alternatively, click Configure next to the Rule text box and use the Create Expression dialog box to create an expression.
6. Click OK.
7. Verify that the virtual server for which you configured persistence is correctly configured by selecting the virtual server and viewing the Details section at the bottom of the pane.

Configuring Features to Protect the Load Balancing Configuration

Jun 24, 2013

You can configure URL redirection to provide notifications of virtual server malfunctions, and you can configure backup virtual servers to take over if a primary virtual server becomes unavailable.

Updated: 2013-06-24

You can configure a redirect URL to communicate the status of the appliance in the event that a virtual server of type HTTP or HTTPS is down or disabled. This URL can be a local or remote link. The appliance uses HTTP 302 redirect.

Redirects can be absolute URLs or relative URLs. If the configured redirect URL contains an absolute URL, the HTTP redirect is sent to the configured location, regardless of the URL specified in the incoming HTTP request. If the configured redirect URL contains only the domain name (relative URL), the HTTP redirect is sent to a location after appending the incoming URL to the domain configured in the redirect URL.

Note: If a load balancing virtual server is configured with both a backup virtual server and a redirect URL, the backup virtual server takes precedence over the redirect URL. In this case, a redirect is used when both the primary and backup virtual servers are down.

To configure a virtual server to redirect client requests to a URL by using the command line interface

At the command prompt, type the following commands to configure a virtual server to redirect client requests to a URL and verify the configuration:

- set lb vserver <name> -redirectURL <URL>
- show lb vserver <name>

Example

```
> set lb vserver vserver-LB-1 -redirectURL http://www.newdomain.com/mysite/maintenance
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   State: DOWN
   Last state change was at Wed Jun 17 08:56:34 2009 (+666 ms)
   .
   .
   .
   Redirect URL: http://www.newdomain.com/mysite/maintenance
   .
   .
   .
 Done
>
```

To configure a virtual server to redirect client requests to a URL by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure URL redirection (for example, vserver-LB-1), and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, in the Redirect URL text box, type the URL (for example, http://www.newdomain.com/mysite/maintenance), and then click OK.
4. Verify that the redirect URL you configured for the server appears in the Details section at the bottom of the pane.

Updated: 2013-06-24

If the primary virtual server is down or disabled, the appliance can direct the connections or client requests to a backup virtual server that forwards the client traffic to the services. The appliance can also send a notification message to the client regarding the site outage or maintenance. The backup virtual server is a proxy and is transparent to the client.

You can configure a backup virtual server when you create a virtual server or when you change the optional parameters of an existing virtual server. You can also configure a backup virtual server for an existing backup virtual server, thus creating a cascaded backup virtual server. The maximum depth of cascading backup virtual servers is 10. The appliance searches for a backup virtual server that is up and accesses that virtual server to deliver the content.

You can configure URL redirection on the primary for use when the primary and the backup virtual servers are down or have reached their thresholds for handling requests.

Note: If no backup virtual server exists, an error message appears, unless the virtual server is configured with a redirect URL. If both a backup virtual server and a redirect URL are configured, the backup virtual server takes precedence.

To configure a backup virtual server by using the command line interface

At the command prompt, type the following commands to configure a backup server and verify the configuration:

- set lb vserver <name> [-backupVserver <string>]
- show lb vserver <name>

Example

```
> set lb vserver vserver-LB-1 -backupVserver vserver-LB-2
 Done
> show lb vserver vserver-LB-1
   vserver-LB-1 (10.102.29.60:80) - HTTP   Type: ADDRESS
   State: DOWN
   Last state change was at Wed Jun 17 08:56:34 2009 (+661 ms)
   .
   .
   .
   Backup: vserver-LB-2
   .
   .
```

.
Done
>

To set up a backup virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the backup virtual server (for example, vserver-LB-1), and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, in the Backup Virtual Server list, select the backup virtual server (for example, vserver-LB-2, and then click OK.
4. Verify that the backup virtual server you configured appears in the Details section at the bottom of the pane.
   Note: If the primary server goes down and then comes back up, and you want the backup virtual server to function as the primary server until you explicitly reestablish the primary virtual server, select the Disable Primary When Down check box.

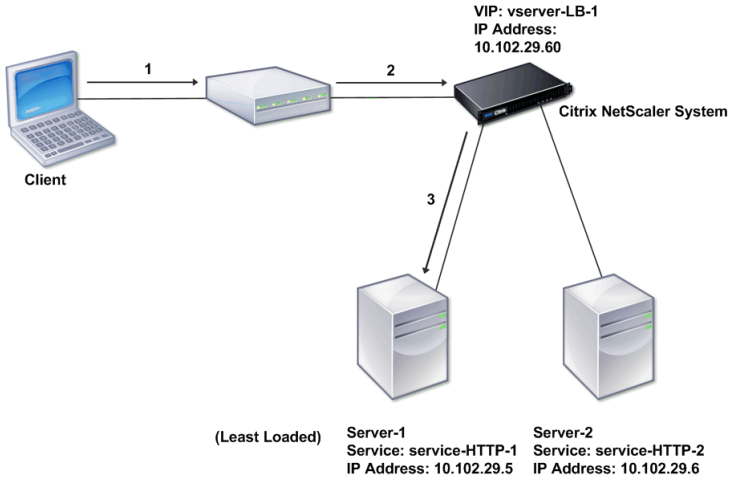
A Typical Load Balancing Scenario

Aug 23, 2013

In a load balancing setup, the NetScaler appliances are logically located between the client and the server farm, and they manage traffic flow to the servers.

The following figure shows the topology of a basic load balancing configuration.

Figure 1. Basic Load Balancing Topology



The virtual server selects the service and assigns it to serve client requests. Consider the scenario in the preceding figure, where the services service-HTTP-1 and service-HTTP-2 are created and bound to the virtual server named virtual server-LB-1. Virtual server-LB-1 forwards the client request to either service-HTTP-1 or service-HTTP-2. The system selects the service for each request by using the Least Connections load balancing method. The following table lists the names and values of the basic entities that must be configured on the system.

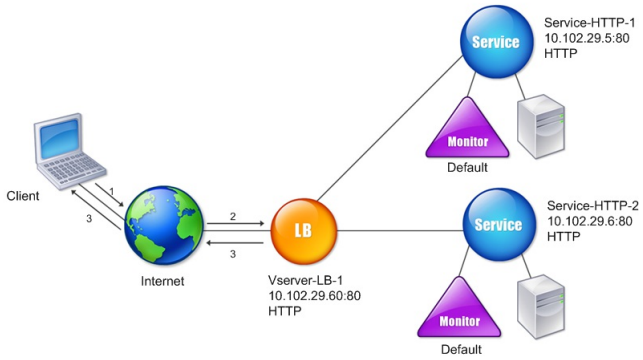
Table 1. LB Configuration Parameter Values

Entity Type	Required parameters and sample values			
	Name	**IP Address**	**Port**	**Protocol**
Virtual Server	vserver-LB-1	10.102.29.60	80	HTTP
Services	service-HTTP-1	10.102.29.5	8083	HTTP
	service-HTTP-2	10.102.29.6	80	HTTP
Monitors	Default	None	None	None

Entity Type	Required parameters and sample values			
	Name	IP Address	Port	Protocol

The following figure shows the load balancing sample values and required parameters that are described in the preceding table.

Figure 2. Load Balancing Entity Model



The following tables list the commands used to configure this load balancing setup by using the command line interface.

Table 2. Initial Configuration Tasks

Task	Command
To enable load balancing	enable feature lb
To create a service named service-HTTP-1	add service service-HTTP-1 10.102.29.5 HTTP 80
To create a service named service-HTTP-2	add service service-HTTP-2 10.102.29.6 HTTP 80
To create a virtual server named vserver-LB-1	add lb vserver vserver-LB-1 HTTP 10.102.29.60 80
To bind a service named service-HTTP-1 to a virtual server named vserver-LB-1	bind lb vserver vserver-LB-1 service-HTTP-1
To bind a service named service-HTTP-2 to a virtual server named vserver-LB-1	bind lb vserver vserver-LB-1 service-HTTP-2

For more information about the initial configuration tasks, see "Enabling Load Balancing" and "Configuring Services and a Vserver."

Table 3. Verification Tasks

Task	Command
To view the properties of a virtual server named vserver-LB-1	show lb vserver vserver-LB-1
To view the statistics of a virtual server named vserver-LB-1	stat lb vserver vserver-LB-1
To view the properties of a service named service-HTTP-1	show service service-HTTP-1
To view the statistics of a service named service-HTTP-1	stat service service-HTTP-1
To view the bindings of a service named service-HTTP-1	show service bindings service-HTTP-1

Table 4. Customization Tasks

Task	Command
To configure persistence on a virtual server named vserver-LB-1	set lb vserver vserver-LB-1 -persistenceType SOURCEIP -persistenceMask 255.255.255.255 -timeout 2
To configure COOKIEINSERT persistence on a virtual server named vserver-LB-1	set lb vserver vserver-LB-1 -persistenceType COOKIEINSERT
To configure URLPassive persistence on a virtual server named vserver-LB-1	set lb vserver vserver-LB-1 -persistenceType URLPASSIVE
To configure a virtual server to redirect the client request to a URL on a virtual server named vserver-LB-1	set lb vserver vserver-LB-1 -redirectURL http://www.newdomain.com/mysite/maintenance
To set a backup virtual server on a virtual server named vserver-LB-1	set lb vserver vserver-LB-1 -backupVserver vserver-LB-2

For more information about configuring persistence, see "Choosing and Configuring Persistence Settings." For information about configuring a virtual server to redirect a client request to a URL and setting up a backup virtual server, see "Configuring Features to Protect the Load Balancing Configuration."

Accelerating Load Balanced Traffic by Using Compression

Aug 22, 2013

Compression is a popular means of optimizing bandwidth usage, and most web browsers support compressed data. If you enable the compression feature, the NetScaler appliance intercepts requests from clients and determines whether the client can accept compressed content. After receiving the HTTP response from the server, the appliance examines the content to determine whether it is compressible. If the content is compressible, the appliance compresses it, modifies the response header to indicate the type of compression performed, and forwards the compressed content to the client.

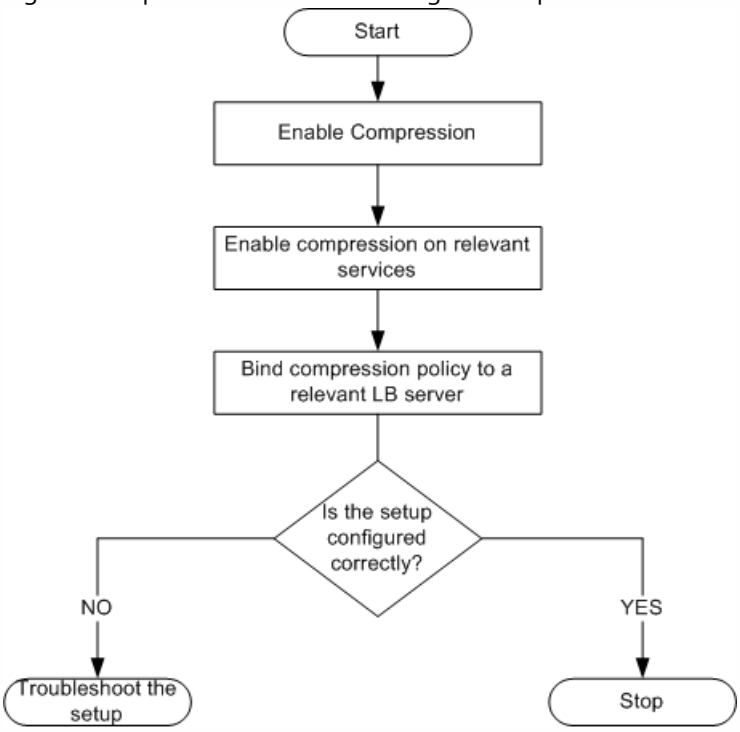
NetScaler compression is a policy-based feature. A policy filters requests and responses to identify responses to be compressed, and specifies the type of compression to apply to each response. The appliance provides several built-in policies to compress common MIME types such as text/html, text/plain, text/xml, text/css, text/rtf, application/msword, application/vnd.ms-excel, and application/vnd.ms-powerpoint. You can also create custom policies. The appliance does not compress compressed MIME types such as application/octet-stream, binary, bytes, and compressed image formats such as GIF and JPEG.

To configure compression, you must enable it globally and on each service that will provide responses that you want compressed. If you have configured virtual servers for load balancing or content switching, you should bind the polices to the virtual servers. Otherwise, the policies apply to all traffic that passes through the appliance.

Updated: 2013-08-22

The following flow chart shows the sequence of tasks for configuring basic compression in a load balancing setup.

Figure 1. Sequence of Tasks to Configure Compression



Note: The steps in the above figure assume that load balancing has already been configured.

Updated: 2013-06-07

By default, compression is not enabled. You must enable the compression feature to allow compression of HTTP responses that are sent to the client.

To enable compression by using the command line interface

At the command prompt, type the following commands to enable compression and verify the configuration:

- enable ns feature CMP
- show ns feature

Example

```
> enable ns feature CMP
Done
> show ns feature

     Feature                Acronym          Status
     -------                -------          ------
1)   Web Logging             WL               ON
2)   Surge Protection        SP               OFF
.
7)   Compression Control     CMP              ON
8)   Priority Queuing        PQ               OFF
.
Done
```

To enable compression by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Compression check box, and then click OK.
4. In the Enable/Disable Feature(s)? dialog box, click Yes.

Updated: 2013-08-22

In addition to enabling compression globally, you must enable it on each service that will deliver files to be compressed.

To enable compression on a service by using the command line

At the command prompt, type the following commands to enable compression on a service and verify the configuration:

- set service <name> -CMP YES
- show service <name>

Example


```
> show service SVC_HTTP1
SVC_HTTP1 (10.102.29.18:80) - HTTP
State: UP
Last state change was at Tue Jun 16 06:19:14 2009 (+737 ms)
Time since last state change: 0 days, 03:03:37.200
Server Name: 10.102.29.18
Server ID : 0   Monitor Threshold : 0
Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
Use Source IP: NO
Client Keepalive(CKA): NO
Access Down Service: NO
TCP Buffering(TCPB): NO
HTTP Compression(CMP): YES
Idle timeout: Client: 180 sec   Server: 360 sec
Client IP: DISABLED
Cacheable: NO
SC: OFF
SP: OFF
Down state flush: ENABLED

1)      Monitor Name: tcp-default
State: DOWN     Weight: 1
Probes: 1095    Failed [Total: 1095 Current: 1095]
Last response: Failure - TCP syn sent, reset received.
Response Time: N/A
Done
```

To enable compression on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure compression (for example, service-HTTP-1), and then click Open.
3. On the Advanced tab, under Settings, select the Compression check box, and then click OK.
4. Verify that, when the service is selected, HTTP Compression(CMP): ON appears in the **Details** section at the bottom of the pane.

Updated: 2013-09-04

If you bind a policy to a virtual server, the policy is evaluated only by the services associated with that virtual server. You can bind compression policies to a virtual server either from the Configure Virtual Server (Load Balancing) dialog box or from the Compression Policy Manager dialog box. This topic includes instructions to bind compression policies to a load balancing virtual server by using the Configure Virtual Server (Load Balancing) dialog box. For information about how you can bind a compression policy to a load balancing virtual server by using the Compression Policy Manager dialog box, see "Configuring and Binding Policies with the Policy Manager."

To bind or unbind a compression policy to a virtual server by using the command line

At the command prompt, type the following commands to bind or unbind a compression policy to a load balancing virtual server and verify the configuration:

- (bind | unbind) lb vserver <name> -policyName <string>
- show lb vserver <name>

Example

```
> bind lb vserver lbvip -policyName ns_cmp_msapp
Done
> show lb vserver lbvip
lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
State: UP
Last state change was at Thu May 28 05:37:21 2009 (+685 ms)
Time since last state change: 19 days, 04:26:50.470
Effective State: UP
Client Idle Timeout: 180 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
Port Rewrite : DISABLED
No. of Bound Services :  1 (Total)      1 (Active)
Configured Method: LEASTCONNECTION
Current Method: Round Robin, Reason: Bound service's state changed to UP
Mode: IP
Persistence: NONE
Vserver IP and Port insertion: OFF
Push: DISABLED  Push VServer:
Push Multi Clients: NO
Push Label Rule:

Bound Service Groups:
1)      Group Name: Service-Group-1

1) Service-Group-1 (10.102.29.252: 80) - HTTP State: UP Weight:                 1

1)      Policy : ns_cmp_msapp Priority:0
Done
```

To bind or unbind a compression policy to a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind or unbind a compression policy (for example, Vserver-LB-1), and then click Open.

3. In the Configure Virtual Server (Load Balancing) dialog box, on the Policies tab, click Compression.
4. Do one of the following:
   - To bind a compression policy, click Insert Policy, and then select the policy you want to bind to the virtual server.
   - To unbind a compression policy, click the name of the policy you want to unbind from the virtual server, and then click Unbind Policy.
5. Click OK.

Securing Load Balanced Traffic by Using SSL

Jan 31, 2011

The Citrix NetScaler SSL offload feature transparently improves the performance of web sites that conduct SSL transactions. By offloading CPU-intensive SSL encryption and decryption tasks from the local web server to the appliance, SSL offloading ensures secure delivery of web applications without the performance penalty incurred when the server processes the SSL data. Once the SSL traffic is decrypted, it can be processed by all standard services. The SSL protocol works seamlessly with various types of HTTP and TCP data and provides a secure channel for transactions using such data.

To configure SSL, you must first enable it. Then, you configure HTTP or TCP services and an SSL virtual server on the appliance, and bind the services to the virtual server. You must also add a certificate-key pair and bind it to the SSL virtual server. If you use Outlook Web Access servers, you must create an action to enable SSL support and a policy to apply the action. An SSL virtual server intercepts incoming encrypted traffic and decrypts it by using a negotiated algorithm. The SSL virtual server then forwards the decrypted data to the other entities on the appliance for appropriate processing.

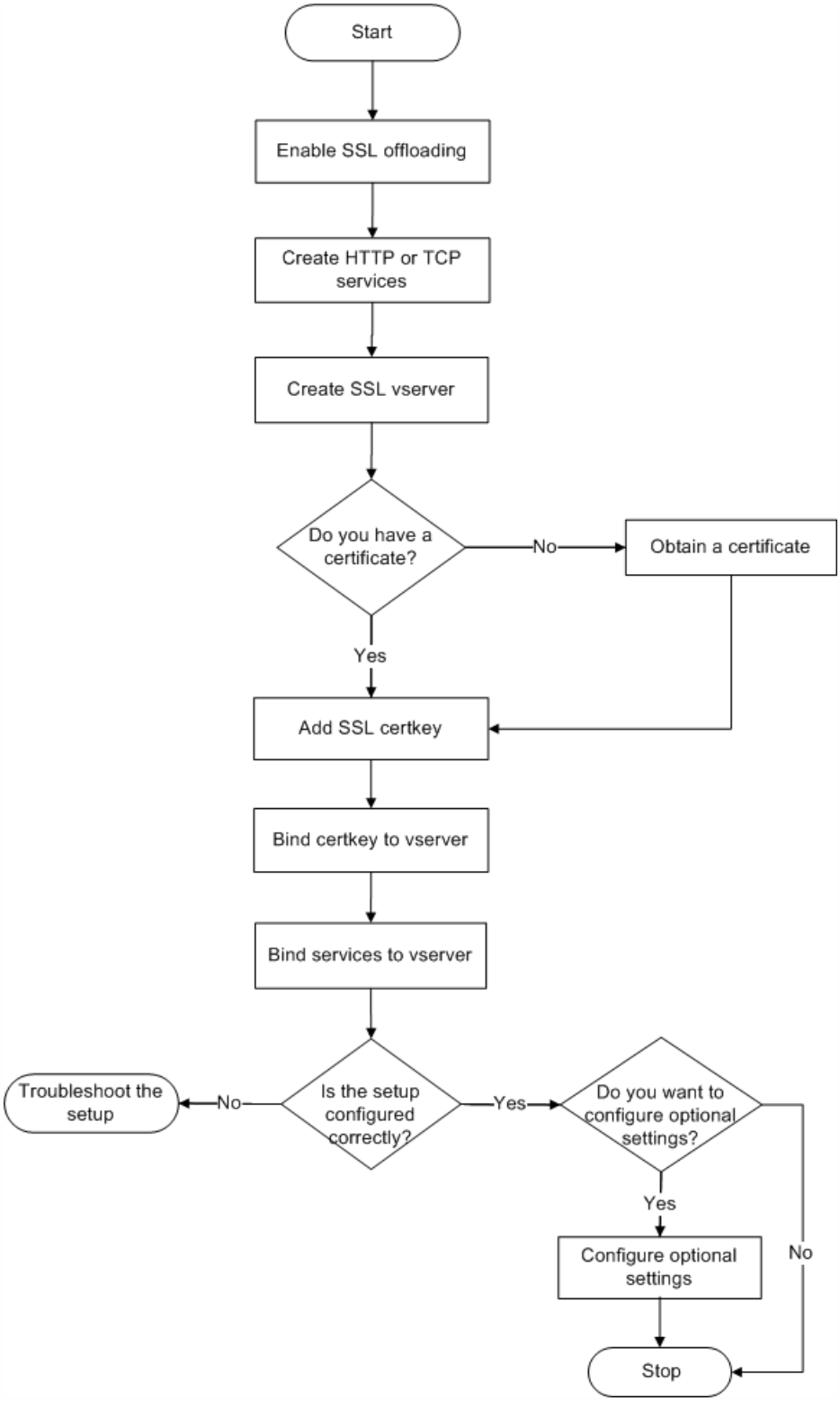
This document includes the following:

To configure SSL, you must first enable it. Then, you must create an SSL virtual server and HTTP or TCP services on the NetScaler. Finally, you must bind a valid SSL certificate and the configured services to the SSL virtual server.

An SSL virtual server intercepts incoming encrypted traffic and decrypts it using a negotiated algorithm. The SSL virtual server then forwards the decrypted data to the other entities on the NetScaler for appropriate processing.

The following flow chart shows the sequence of tasks for configuring a basic SSL offload setup.

Figure 1. Sequence of Tasks to Configure SSL Offloading



Updated: 2013-06-05

You should enable the SSL feature before configuring SSL offload. You can configure SSL-based entities on the appliance without enabling the SSL feature, but they will not work until you enable SSL.

To enable SSL by using the command line interface

At the command prompt, type the following commands to enable SSL Offload and verify the configuration:

- enable ns feature SSL
- show ns feature

Example

```
> enable ns feature ssl
Done
> show ns feature
Feature Acronym Status
------- ------- ------
1) Web Logging WL ON
2) SurgeProtection SP OFF
3) Load Balancing LB ON . . .
 9) SSL Offloading SSL ON
10) Global Server Load Balancing GSLB ON . .
Done >
```

To enable SSL by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. Select the SSL Offloading check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes.

Updated: 2013-08-23

A service on the appliance represents an application on a server. Once configured, services are in the disabled state until the appliance can reach the server on the network and monitor its status. This topic covers the steps to create an HTTP service.

Note: For TCP traffic, perform the procedures in this and the following topics, but create TCP services instead of HTTP services.

To add an HTTP service by using the command line interface

At the command prompt, type the following commands to add a HTTP service and verify the configuration:

- add service <name> (<IP> | <serverName>) <serviceType> <port>
- show service <name>

```
> add service SVC_HTTP1 10.102.29.18 HTTP 80
 Done
> show service SVC_HTTP1
      SVC_HTTP1 (10.102.29.18:80) - HTTP
      State: UP
      Last state change was at Wed Jul 15 06:13:05 2009
```



```
        Time since last state change: 0 days, 00:00:15.350
        Server Name: 10.102.29.18
        Server ID : 0   Monitor Threshold : 0
        Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
        Use Source IP: NO
        Client Keepalive(CKA): NO
        Access Down Service: NO
        TCP Buffering(TCPB): NO
        HTTP Compression(CMP): YES
        Idle timeout: Client: 180 sec   Server: 360 sec
        Client IP: DISABLED
        Cacheable: NO
        SC: OFF
        SP: OFF
        Down state flush: ENABLED

    1)    Monitor Name: tcp-default
              State: UP       Weight: 1
              Probes: 4       Failed [Total: 0 Current: 0]
              Last response: Success - TCP syn+ack received.
              Response Time: N/A
    Done
```

To add an HTTP service by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Services.
2. In details pane, click Add.
3. In the Create Service dialog box, in the Service Name, Server, and Port text boxes, type the name of the service, IP address, and port (for example, SVC_HTTP1, 10.102.29.18, and 80).
4. In the Protocol list, select the type of the service (for example, HTTP).
5. Click Create, and then click Close. The HTTP service you configured appears in the Services page.
6. Verify that the parameters you configured are correctly configured by selecting the service and viewing the Details section at the bottom of the pane.

Updated: 2013-06-05

In a basic SSL offloading setup, the SSL virtual server intercepts encrypted traffic, decrypts it, and sends the clear text messages to the services that are bound to the virtual server. Offloading CPU-intensive SSL processing to the appliance allows the back-end servers to process a greater number of requests.

To add an SSL-based virtual server by using the command line interface

At the command prompt, type the following commands to create an SSL-based virtual server and verify the configuration:

- add lb vserver <name> <serviceType> [<IPAddress> <port>]
- show lb vserver <name>

Example

```
> add lb vserver vserver-SSL-1 SSL 10.102.29.50 443
Done
> show lb vserver vserver-SSL-1
vserver-SSL-1 (10.102.29.50:443) - SSL Type: ADDRESS
State: DOWN[Certkey not bound] Last state change was at Tue Jun 16 06:33:08 2009 (+176 ms)
Time since last state change: 0 days, 00:03:44.120
Effective State: DOWN Client Idle Timeout: 180 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
No. of Bound Services : 0 (Total) 0 (Active)
Configured Method: LEASTCONNECTION Mode: IP
Persistence: NONE
Vserver IP and Port insertion: OFF
Push: DISABLED Push VServer: Push Multi Clients: NO Push Label Rule: Done
```

Caution: To ensure secure connections, you must bind a valid SSL certificate to the SSL-based virtual server before you enable it.

To add an SSL-based virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (SSL Offload) dialog box, in the Name, IP Address, and Port text boxes, type the name of the virtual server, IP address, and port (for example, Vserver-SSL-1, 10.102.29.50, and 443).
4. In the Protocol list, select the type of the virtual server, for example, SSL.
5. Click Create, and then click Close.
6. Verify that the parameters you configured are correctly configured by selecting the virtual server and viewing the Details section at the bottom of the pane. The virtual server is marked as DOWN because a certificate-key pair and services have not been bound to it.

Caution: To ensure secure connections, you must bind a valid SSL certificate to the SSL-based virtual server before you enable it.

Updated: 2013-08-23

After decrypting the incoming data, the SSL virtual server forwards the data to the services that you have bound to the virtual server.

Data transfer between the appliance and the servers can be encrypted or in clear text. If the data transfer between the appliance and the servers is encrypted, the entire transaction is secure from end to end. For more information about configuring the system for end-to-end security, see "SSL Offload and Acceleration."

To bind a service to a virtual server by using the command line interface

At the command prompt, type the following commands to bind service to the SSL virtual server and verify the configuration:

- bind lb vserver <name> <serviceName>
- show lb vserver <name>

Example

```
> bind lb vserver vserver-SSL-1 SVC_HTTP1
Done
> show lb vserver vserver-SSL-1 vserver-SSL-1 (10.102.29.50:443) - SSL Type:
ADDRESS State: DOWN[Certkey not bound]
Last state change was at Tue Jun 16 06:33:08 2009 (+174 ms)
Time since last state change: 0 days, 00:31:53.70
Effective State: DOWN Client Idle
Timeout: 180 sec
Down state flush: ENABLED Disable Primary Vserver On Down :
DISABLED No. of Bound Services : 1 (Total) 0 (Active)
Configured Method: LEASTCONNECTION Mode: IP Persistence: NONE Vserver IP and
Port insertion: OFF Push: DISABLED Push VServer: Push Multi Clients: NO Push Label Rule:

1) SVC_HTTP1 (10.102.29.18: 80) - HTTP
State: DOWN Weight: 1
Done
```

To bind a service to a virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, select a virtual server, and then click Open.
3. On the Services tab, in the Active column, select the check boxes next to the services that you want to bind to the selected virtual server.
4. Click OK.
5. Verify that the Number of Bound Services counter in the Details section at the bottom of the pane is incremented by the number of services that you bound to the virtual server.

Updated: 2013-06-24

An SSL certificate is an integral element of the SSL Key-Exchange and encryption/decryption process. The certificate is used during SSL handshake to establish the identity of the SSL server. You can use a valid, existing SSL certificate that you have on the NetScaler appliance, or you can create your own SSL certificate. The appliance supports RSA/DSA certificates of up to 4096 bits.

Note: Citrix recommends that you use a valid SSL certificate that has been issued by a trusted certificate authority. Invalid certificates and self-created certificates are not compatible with all SSL clients.
Before a certificate can be used for SSL processing, you must pair it with its corresponding key. The certificate key pair is then bound to the virtual server and used for SSL processing.

To add a certificate key pair by using the command line interface

At the command prompt, type the following commands to create a certificate key pair and verify the configuration:

- add ssl certKey <certkeyName> -cert <string> [-key <string>]
- show sslcertkey <name>

Example

```
> add ssl certKey CertKey-SSL-1 -cert ns-root.cert -key ns-root.key
 Done
> show sslcertkey CertKey-SSL-1
   Name: CertKey-SSL-1 Status: Valid,
   Days to expiration:4811 Version: 3
   Serial Number: 00 Signature Algorithm: md5WithRSAEncryption Issuer: C=US,ST=California,L=San
   Jose,O=Citrix ANG,OU=NS Internal,CN=de fault
   Validity Not Before: Oct 6 06:52:07 2006 GMT Not After : Aug 17 21:26:47 2022 GMT
   Subject: C=US,ST=California,L=San Jose,O=Citrix ANG,OU=NS Internal,CN=d efault Public Key
   Algorithm: rsaEncryption Public Key
   size: 1024
 Done
```

To add a certificate key pair by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. In the details pane, click Add.
3. In the Install Certificate dialog box, in the Certificate-Key Pair Name text box, type a name for the certificate key pair you want to add, for example, Certkey-SSL-1.
4. Under Details, in Certificate File Name, click Browse (Appliance) to locate the certificate. Both the certificate and the key are stored in the /nsconfig/ssl/ folder on the appliance. To use a certificate present on the local system, select Local.
5. Select the certificate you want to use, and then click Select.
6. In Private Key File Name, click Browse (Appliance) to locate the private key file. To use a private key present on the local system, select Local.
7. Select the key you want to use and click Select. To encrypt the key used in the certificate key pair, type the password to be used for encryption in the Password text box.
8. Click Install.
9. Double-click the certificate key pair and, in the Certificate Details window, verify that the parameters have been configured correctly and saved.

Updated: 2013-06-24

After you have paired an SSL certificate with its corresponding key, you must bind the certificate key pair to the SSL virtual server so that it can be used for SSL processing. Secure sessions require establishing a connection between the client computer and an SSL-based virtual server on the appliance. SSL processing is then carried out on the incoming traffic at the virtual server. Therefore, before enabling the SSL virtual server on the appliance, you need to bind a valid SSL certificate to the SSL virtual server.

To bind an SSL certificate key pair to a virtual server by using the command line

interface

At the command prompt, type the following commands to bind an SSL certificate key pair to a virtual server and verify the configuration:

- bind ssl vserver <vServerName> -certkeyName <string>
- show ssl vserver <name>

Example

```
 > bind ssl vserver Vserver-SSL-1 -certkeyName CertKey-SSL-1
Done
> show ssl vserver Vserver-SSL-1

    Advanced SSL configuration for VServer Vserver-SSL-1:
    DH: DISABLED
    Ephemeral RSA: ENABLED Refresh Count: 0
    Session Reuse: ENABLED Timeout: 120 seconds
    Cipher Redirect: ENABLED
    SSLv2 Redirect: ENABLED
    ClearText Port: 0
    Client Auth: DISABLED
    SSL Redirect: DISABLED
    Non FIPS Ciphers: DISABLED
    SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED

1) CertKey Name: CertKey-SSL-1 Server Certificate
1) Cipher Name: DEFAULT
  Description: Predefined Cipher Alias
Done
```

To bind an SSL certificate key pair to a virtual server by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server to which you want to bind the certificate key pair, for example, Vserver-SSL-1, and click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, on the SSL Settings tab, under Available, select the certificate key pair that you want to bind to the virtual server (for example, Certkey-SSL-1), and then click Add.
4. Click OK.
5. Verify that the certificate key pair that you selected appears in the Configured area.

If you use Outlook Web Access (OWA) servers on your NetScaler appliance, you must configure the appliance to insert a special header field, FRONT-END-HTTPS: ON, in HTTP requests directed to the OWA servers, so that the servers generate URL links as https:// instead of http://.

Note: You can enable OWA support for HTTP-based SSL virtual servers and services only. You cannot apply it for TCP-based SSL virtual servers and services.
To configure OWA support, do the following:

- Create an SSL action to enable OWA support.
- Create an SSL policy.
- Bind the policy to the SSL virtual server.

Creating an SSL Action to Enable OWA Support

Updated: 2013-06-24

Before you can enable Outlook Web Access (OWA) support, you must create an SSL action. SSL actions are bound to SSL policies and triggered when incoming data matches the rule specified by the policy.

At the command prompt, type the following commands to create an SSL action to enable OWA support and verify the configuration:

- add ssl action <name> -OWASupport ENABLED
- show SSL action <name>

```
 > add ssl action Action-SSL-OWA -OWASupport enabled
Done
> show SSL action Action-SSL-OWA
  Name: Action-SSL-OWA
  Data Insertion Action: OWA
  Support: ENABLED
Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, on the Actions tab, click Add.
3. In the Create SSL Action dialog box, in the Name text box, type Action-SSL-OWA.
4. Under Outlook Web Access, select Enabled.
5. Click Create, and then click Close.
6. Verify that Action-SSL-OWA appears in the **SSL Actions** page.

Creating SSL Policies

Updated: 2013-09-04

SSL policies are created by using the policy infrastructure. Each SSL policy has an SSL action bound to it, and the action is carried out when incoming traffic matches the rule that has been configured in the policy.

At the command prompt, type the following commands to configure an SSL policy and verify the configuration:

- add ssl policy <name> -rule <expression> -reqAction <string>
- show ssl policy <name>

Example

```
> add ssl policy Policy-SSL-1 -rule ns_true -reqaction Action-SSL-OWA
```



```
Done
> show ssl policy Policy-SSL-1
Name: Policy-SSL-1      Rule: ns_true
Action: Action-SSL-OWA  Hits: 0
Policy is bound to following entities
1)      PRIORITY : 0
Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, click Add.
3. In the Create SSL Policy dialog box, in the Name text box, type the name of the SSL Policy (for example, Policy-SSL-1).
4. In Request Action, select the configured SSL action that you want to associate with this policy (for example, Action-SSL-OWA). The ns_true general expression applies the policy to all successful SSL handshake traffic. However, if you need to filter specific responses, you can create policies with a higher level of detail. For more information about configuring granular policy expressions, see "Understanding Policies and Expressions."
5. In Named Expressions, choose the built-in general expression ns_true and click Add Expression. The expression ns_true now appears in the Expression text box.
6. Click Create, and then click Close.
7. Verify that the policy is correctly configured by selecting the policy and viewing the Details section at the bottom of the pane.

Binding the SSL Policy to an SSL Virtual Server

Updated: 2013-06-24

After you configure an SSL policy for Outlook Web Access, bind the policy to a virtual server that will intercept incoming Outlook traffic. If the incoming data matches any of the rules configured in the SSL policy, the policy is triggered and the action associated with it is carried out.

At the command prompt, type the following commands to bind an SSL policy to an SSL virtual server and verify the configuration:

- bind ssl vserver <vServerName> -policyName <string>
- show ssl vserver <name>

Example

```
> bind ssl vserver Vserver-SSL-1 -policyName Policy-SSL-1
 Done
> show ssl vserver Vserver-SSL-1
        Advanced SSL configuration for VServer Vserver-SSL-1:
        DH: DISABLED
        Ephemeral RSA: ENABLED          Refresh Count: 0
        Session Reuse: ENABLED          Timeout: 120 seconds
        Cipher Redirect: ENABLED
```

```
        SSLv2 Redirect: ENABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED

    1)    CertKey Name: CertKey-SSL-1 Server Certificate

    1)    Policy Name: Policy-SSL-1
                    Priority: 0
    1)    Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
     Done
    >
```

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, select the virtual server (for example, Vserver-SSL-1), and then click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, click Insert Policy, and then select the policy that you want to bind to the SSL virtual server. Optionally, you can double-click the Priority field and type a new priority level.
4. Click OK.

Features at a Glance

Sep 04, 2013

Citrix NetScaler features can be configured independently or in combinations to address specific needs. Although some features fit more than one category, the numerous NetScaler features can generally be categorized as application switching and traffic management features, application acceleration features, and application security and firewall features, and an application visibility feature.

To understand the order in which the features perform their processing, see "Processing Order of Features."

This document includes the following:

Application Switching and Traffic Management Features

Aug 30, 2016

SSL Offloading

Transparently offloads SSL encryption and decryption from web servers, freeing server resources to service content requests. SSL places a heavy burden on an application's performance and can render many optimization measures ineffective. SSL offload and acceleration allow all the benefits of Citrix Request Switching technology to be applied to SSL traffic, ensuring secure delivery of web applications without degrading end-user performance.
For more information, see "SSL Offload and Acceleration."

Access Control Lists

Compares incoming packets to Access Control Lists (ACLs). If a packet matches an ACL rule, the action specified in the rule is applied to the packet. Otherwise, the default action (ALLOW) is applied and the packet is processed normally. For the appliance to compare incoming packets to the ACLs, you have to apply the ACLs. All ACLs are enabled by default, but you have to apply them in order for the NetScaler to compare incoming packets against them. If an ACL is not required to be a part of the lookup table, but still needs to be retained in the configuration, it should be disabled before the ACLs are applied. A NetScaler does not compare incoming packets to disabled ACLs.
For more information, see "Access Control List."

Load Balancing

Load balancing decisions are based on a variety of algorithms, including round robin, least connections, weighted least bandwidth, weighted least packets, minimum response time, and hashing based on URL, domain source IP, or destination IP. Both the TCP and UDP protocols are supported, so the NetScaler can load balance all traffic that uses those protocols as the underlying carrier (for example, HTTP, HTTPS, UDP, DNS, NNTP, and general firewall traffic). In addition, the NetScaler can maintain session persistence based on source IP, cookie, server, group, or SSL session. It allows users to apply custom Extended Content Verification (ECV) to servers, caches, firewalls and other infrastructure devices to ensure that these systems are functioning properly and are providing the right content to users. It can also perform health checks using ping, TCP, or HTTP URL, and the user can create monitors based on Perl scripts. To provide high-scale WAN optimization, the CloudBridge appliances deployed at data centers can be load balanced through NetScaler appliances. The bandwidth and number of concurrent sessions can be improved significantly.
For more information, see "Load Balancing."

Traffic Domains

Traffic domains provide a way to create logical ADC partitions within a single NetScaler appliance. They enable you to segment network traffic for different applications. You can use traffic domains to create multiple isolated environments whose resources do not interact with each other. An application belonging to a specific traffic domain communicates only with entities, and processes traffic, within that domain. Traffic belonging to one traffic domain cannot cross the boundary of another traffic domain. Therefore, you can use duplicate IP addresses on the appliance as long as an addresses is not duplicated within the same domain.
For more information, see "Traffic Domains."

Network Address Translation

Network address translation (NAT) involves modification of the source and/or destination IP addresses, and/or the TCP/UDP port numbers, of IP packets that pass through the NetScaler appliance. Enabling NAT on the appliance enhances

the security of your private network, and protects it from a public network such as the Internet, by modifying your network's source IP addresses when data passes through the NetScaler.

The NetScaler appliance supports the following types of network address translation:

INAT—In Inbound NAT (INAT), an IP address (usually public) configured on the NetScaler appliance listens to connection requests on behalf of a server. For a request packet received by the appliance on a public IP address, the NetScaler replaces the destination IP address with the private IP address of the server. In other words, the appliance acts as a proxy between clients and the server. INAT configuration involves INAT rules, which define a 1:1 relationship between the IP address on the NetScaler appliance and the IP address of the server.

RNAT—In Reverse Network Address Translation (RNAT), for a session initiated by a server, the NetScaler appliance replaces the source IP address in the packets generated by the server with an IP address (type SNIP) configured on the appliance. The appliance thereby prevents exposure of the server's IP address in any of the packets generated by the server. An RNAT configuration involves an RNAT rule, which specifies a condition. The appliance performs RNAT processing on those packets that match the condition.

Stateless NAT46 Translation—Stateless NAT46 enables communication between IPv4 and IPv6 networks, by way of IPv4 to IPv6 packet translation and vice versa, without maintaining any session information on the NetScaler appliance. A stateless NAT46 configuration involves an IPv4-IPv6 INAT rule and an NAT46 IPv6 prefix.

Stateful NAT64 Translation—The stateful NAT64 feature enables communication between IPv4 clients and IPv6 servers through IPv6 to IPv4 packet translation, and vice versa, while maintaining session information on the NetScaler appliance. A stateful NAT64 configuration involves an NAT64 rule and an NAT64 IPv6 prefix.

For more information, see "Configuring Network Address Translation."

Multipath TCP Support

NetScaler appliances support Multipath TCP (MPTCP). MPTCP is a TCP/IP protocol extension that identifies and uses multiple paths available between hosts to maintain the TCP session. You must enable MPTCP on a TCP profile and bind it to a virtual server. When MPTCP is enabled, the virtual server functions as an MPTCP gateway and converts MPTCP connections with the clients to TCP connections that it maintains with the servers.
For more information, see "MPTCP (Multi-Path TCP)."

Content Switching

Determines the server to which to send the request on the basis of configured content switching policies. Policy rules can be based on the IP address, URL, and HTTP headers. This allows switching decisions to be based on user and device characteristics such as who the user is, what type of agent is being used, and what content the user requested.
For more information, see "Content Switching."

Global Server Load Balancing (GSLB)

Extends the traffic management capabilities of a NetScaler to include distributed Internet sites and global enterprises. Whether installations are spread across multiple network locations or multiple clusters in a single location, the NetScaler maintains availability and distributes traffic across them. It makes intelligent DNS decisions to prevent users from being sent to a site that is down or overloaded. When the proximity-based GSLB method is enabled, the NetScaler can make load balancing decisions based on the proximity of the client's local DNS server (LDNS) in relation to different sites. The main benefit of the proximity-based GSLB method is faster response time resulting from the selection of the closest available site.
For more information, see "Global Server Load Balancing."

Dynamic Routing

Enables routers to obtain topology information, routes, and IP addresses from neighboring routers automatically. When dynamic routing is enabled, the corresponding routing process listens to route updates and advertises routes. The routing processes can also be placed in passive mode. Routing protocols enable an upstream router to load balance traffic to identical virtual servers hosted on two standalone NetScaler units using the Equal Cost Multipath technique.
For more information, see "Configuring Dynamic Routes."

Link Load Balancing

Load balances multiple WAN links and provides link failover, further optimizing network performance and ensuring business continuity. Ensures that network connections remain highly available, by applying intelligent traffic control and health checks to distribute traffic efficiently across upstream routers. Identifies the best WAN link to route both incoming and outbound traffic based on policies and network conditions, and protects applications against WAN or Internet link failure by providing rapid fault detection and failover.
For more information, see " Link Load Balancing."

TCP Optimization

You can use TCP profiles to optimize TCP traffic. TCP profiles define the way that NetScaler virtual servers process TCP traffic. Administrators can use the built-in TCP profiles or configure custom profiles. After defining a TCP profile, you can bind it to a single virtual server or to multiple virtual servers.
Some of the key optimization features that can be enabled by TCP profiles are:

- TCP keep-alive—Checks the operational status of the peers at specified time intervals to prevent the link from being broken.
- Selective Acknowledgment (SACK)— Improves the performance of data transmission, especially in long fat networks (LFNs).
- TCP window scaling— Allows efficient transfer of data over long fat networks (LFNs).

For more information on TCP Profiles, see "Configuring TCP Profiles."

Web Interface on NetScaler

Provides access to XenApp and XenDesktop resources, which include applications, content, and desktops. Users access resources through a standard Web browser or by using the Citrix XenApp plug-in. The Web Interface runs as a service on port 8080 on the NetScaler appliance. To create Web Interface sites, Java is executed on Apache Tomcat Web server version 6.0.26 on the NetScaler appliance.
Note: Web Interface is supported only on NetScaler nCore releases.
For more information, see "Web Interface."

CloudBridge Connector

The Citrix NetScaler CloudBridge Connector feature, a fundamental part of the Citrix OpenCloud framework, is a tool used to build a cloud-extended data center. The OpenCloud Bridge enables you to connect one or more NetScaler appliances or NetScaler virtual appliances on the cloud-to your network without reconfiguring your network. Cloud hosted applications appear as though they are running on one contiguous enterprise network. The primary purpose of the OpenCloud Bridge is to enable companies to move their applications to the cloud while reducing costs and the risk of application failure. In addition, the OpenCloud Bridge increases network security in cloud environments. An OpenCloud Bridge is a Layer-2 network bridge that connects a NetScaler appliance or NetScaler virtual appliance on a cloud instance to a NetScaler appliance or NetScaler virtual appliance on your LAN. The connection is made through a tunnel that uses the Generic Routing Encapsulation (GRE) protocol. The GRE protocol provides a mechanism for encapsulating packets from a wide variety of network protocols to be forwarded over another protocol. Then Internet Protocol security (IPsec) protocol suite is used to secure the communication between the peers in the OpenCloud Bridge.

For more information, see "CloudBridge."

DataStream

The NetScaler DataStream feature provides an intelligent mechanism for request switching at the database layer by distributing requests on the basis of the SQL query being sent.

When deployed in front of database servers, a NetScaler ensures optimal distribution of traffic from the application servers and Web servers. Administrators can segment traffic according to information in the SQL query and on the basis of database names, user names, character sets, and packet size.

You can configure load balancing to switch requests according to load balancing algorithms, or you can elaborate the switching criteria by configuring content switching to make a decision based on SQL query parameters, such as user name, database names, and command parameters. You can further configure monitors to track the states of database servers.

The advanced policy infrastructure on the NetScaler appliance includes expressions that you can use to evaluate and process the requests. The advanced expressions evaluate traffic associated with MySQL database servers. You can use request-based expressions (expressions that begin with MYSQL.CLIENT and MYSQL.REQ) in advanced policies to make request switching decisions at the content switching virtual server bind point and response-based expressions (expressions that begin with MYSQL.RES) to evaluate server responses to user-configured health monitors.

Note: DataStream is supported for MySQL and MS SQL databases.
For more information, see "DataStream."

Application Acceleration Features

Sep 06, 2013

AppCompress

Uses the gzip compression protocol to provide transparent compression for HTML and text files. The typical 4:1 compression ratio yields up to 50% reduction in bandwidth requirements out of the data center. It also results in significantly improved end-user response time, because it reduces the amount of data that must be delivered to the user's browser.

For more information, see "Compression."

Cache Redirection

Manages the flow of traffic to a reverse proxy, transparent proxy, or forward proxy cache farm. Inspects all requests, and identifies non-cacheable requests and sends them directly to the origin servers over persistent connections. By intelligently redirecting non-cacheable requests back to the origin web servers, the NetScaler appliance frees cache resources and increases cache hit rates while reducing overall bandwidth consumption and response delays for these requests.

For more information, see "Cache Redirection."

AppCache

Helps optimize web content and application data delivery by providing a fast in-memory HTTP/1.1 and HTTP/1.0 compliant web caching for both static and dynamic content. This on-board cache stores the results of incoming application requests even when an incoming request is secured or the data compressed, and then reuses the data to fulfill subsequent requests for the same information. By serving data directly from the on-board cache, the appliance can reduce page regeneration times by eliminating the need to funnel static and dynamic content requests to the server.

For more information, see "Integrated Caching."

TCP Buffering

Buffers the server's response and delivers it to the client at the client's speed, thus offloading the server faster and thereby improving the performance of web sites.

For more information, see "TCP Buffering."

Application Security and Firewall Features

Oct 30, 2013

Denial of Service Attack (DoS) Defense

Detects and stops malicious distributed denial-of-service (DDoS) attacks and other types of malicious attacks before they reach your servers, preventing them from affecting network and application performance. The NetScaler appliance identifies legitimate clients and elevates their priority, leaving suspect clients unable to consume a disproportionate percentage of resources and cripple your site. The appliance provides application-level protection from the following types of malicious attacks:

- SYN flood attacks
- Pipeline attacks
- Teardrop attacks
- Land attacks
- Fraggle attacks
- Zombie connection attacks

The appliance aggressively defends against these types of attacks by preventing the allocation of server resources for these connections. This insulates servers from the overwhelming flood of packets associated with these events.

The appliance also protects network resources from ICMP based attacks by using ICMP rate limiting and aggressive ICMP packet inspection. It performs strong IP reassembly, drops a variety of suspicious and malformed packets, and applies Access Control Lists (ACLs) to site traffic for further protection.

For more information, see "HTTP Denial-of-Service Protection."

Content Filtering

Provides protection from malicious attacks for web sites at the Layer 7 level. The appliance inspects each incoming request according to user-configured rules based on HTTP headers, and performs the action the user configured. Actions can include resetting the connection, dropping the request, or sending an error message to the user's browser. This allows the appliance to screen unwanted requests and reduces your servers' exposure to attacks.
This feature can also analyze HTTP GET and POST requests and filter out known bad signatures, allowing it to defend your servers against HTTP-based attacks.

For more information, see "Content Filtering."

Responder

Functions like an advanced filter and can be used to generate responses from the appliance to the client. Some common uses of this feature are generation of redirect responses, user defined responses, and resets.
For more information, see "Responder."

Rewrite

Modifies HTTP headers and body text. You can use the rewrite feature to add HTTP headers to an HTTP request or response, make modifications to individual HTTP headers, or delete HTTP headers. It also enables you to modify the HTTP body in requests and responses.
When the appliance receives a request or sends a response, it checks for rewrite rules, and if applicable rules exist, it applies them to the request or response before passing it on to the web server or client computer.

For more information, see "Rewrite."

Priority Queuing

Prioritizes user requests to ensure that the most important traffic is serviced first during surges in request volume. You can establish priority based on request URLs, cookies, or a variety of other factors. The appliance places requests in a three-tier queue based on their configured priority, enabling business-critical transactions to flow smoothly even during surges or site attacks.
For more information, see "Priority Queuing."

Surge Protection

Regulates the flow of user requests to servers and controls the number of users that can simultaneously access the resources on the servers, queuing any additional requests once your servers have reached their capacity. By controlling the rate at which connections can be established, the appliance blocks surges in requests from being passed on to your servers, thus preventing site overload.
For more information, see "Surge Protection."

NetScaler Gateway

NetScaler Gateway is a secure application access solution that provides administrators granular application-level policy and action controls to secure access to applications and data while allowing users to work from anywhere. It gives IT administrators a single point of control and tools to help ensure compliance with regulations and the highest levels of information security across and outside the enterprise. At the same time, it empowers users with a single point of access—optimized for roles, devices, and networks—to the enterprise applications and data they need. This unique combination of capabilities helps maximize the productivity of today's mobile workforce.

For more information, see "NetScaler Gateway."

Application Firewall

Protects applications from misuse by hackers and malware, such as cross site scripting attacks, buffer overflow attacks, SQL injection attacks, and forceful browsing, by filtering traffic between each protected web server and users that connect to any web site on that web server. The application firewall examines all traffic for evidence of attacks on web server security or misuse of web server resources, and takes the appropriate action to prevent these attacks from succeeding.
For more information, see "Application Firewall."

Application Visibility Feature

Sep 04, 2013

NetScaler Insight Center

NetScaler Insight Center is a high performance collector that provides end-to-end user experience visibility across Web and HDX (ICA) traffic. It collects HTTP and ICA AppFlow records generated by NetScaler ADC appliances and populates analytical reports covering Layer 3 to Layer 7 statistics. NetScaler Insight Center provides in-depth analysis for the last five minutes of real-time data, and for historical data collected for the last one hour, one day, one week, and one month.
HDX (ICA) analytic dashboard enables you to drill down from HDX Users, Applications, Desktops, and even from gateway-level information. Similarly, HTTP analytics provide a bird's eye view of Web Applications, URLs Accessed, Client IP Addresses and Server IP Addresses, and other dashboards. The administrator can drill down and identify the pain points from any of these dashboards, as appropriate for the use case.

EdgeSight for NetScaler

Support for application performance monitoring based on end user experience. This solution leverages the HTML injection feature to obtain various time values, which are used by EdgeSight server for analysis and report generation. EdgeSight for NetScaler provides a way to monitor the performance benefits of a NetScaler and determine potential bottlenecks in a network.
For more information, see "EdgeSight Monitoring for NetScaler."

Enhanced Application Visibility Using AppFlow

The Citrix NetScaler appliance is a central point of control for all application traffic in the data center. It collects flow and user-session level information valuable for application performance monitoring, analytics, and business intelligence applications. AppFlow transmits this information by using the Internet Protocol Flow Information eXport (IPFIX) format, which is an open Internet Engineering Task Force (IETF) standard defined in RFC 5101. IPFIX (the standardized version of Cisco's NetFlow) is widely used to monitor network flow information. AppFlow defines new Information Elements to represent application-level information.
Using UDP as the transport protocol, AppFlow transmits the collected data, called *flow records*, to one or more IPv4 collectors. The collectors aggregate the flow records and generate real-time or historical reports.

AppFlow provides visibility at the transaction level for HTTP, SSL, TCP, and SSL_TCP flows. You can sample and filter the flow types that you want to monitor.

To limit the types of flows to monitor, by sampling and filtering the application traffic, you can enable AppFlow for a virtual server. AppFlow can also provide statistics for the virtual server.

You can also enable AppFlow for a specific service, representing an application server, and monitor the traffic to that application server.

For more information, see "AppFlow."

Stream Analytics

The performance of your web site or application depends on how well you optimize the delivery of the most frequently requested content. Techniques such as caching and compression help accelerate the delivery of services to clients, but you need to be able to identify the resources that are requested most frequently, and then cache or compress those resources. You can identify the most frequently used resources by aggregating real-time statistics about website or application traffic. Statistics such as how frequently a resource is accessed relative to other resources and how much bandwidth is consumed

by those resources help you determine whether those resources need to be cached or compressed to improve server performance and network utilization. Statistics such as response times and the number of concurrent connections to the application help you determine whether you must enhance server-side resources.

If the web site or application does not change frequently, you can use products that collect statistical data, and then manually analyze the statistics and optimize the delivery of content. However, if you do not want to perform manual optimizations, or if your web site or application is dynamic in nature, you need infrastructure that can not only collect statistical data but can also automatically optimize the delivery of resources on the basis of the statistics. On the NetScaler appliance, this functionality is provided by the Stream Analytics feature. The feature operates on a single NetScaler appliance and collects run-time statistics on the basis of criteria that you define. When used with NetScaler policies, the feature also provides you with the infrastructure that you need for automatic, real-time traffic optimization.

For more information, see "Stream Analytics."

Cloud Integration Feature

Aug 23, 2013

AutoScale

All applications have specific usage patterns that comprise peaks and troughs. These load variations can be dynamic in nature and difficult to predict, given that they depend on several factors that are intrinsic to the use case. Cloud users have to constantly monitor the load on their application fleet and make sure that these variations have minimum impact on end users. During periods of peak usage, when the application fleet is overloaded and end users experience significant latency, they have to deploy additional application instances. During trough periods, the expanded fleet is underutilized. So they might have to remove additional instances or bear unnecessary cost overheads. In most cases, they have to perform these tasks manually.

If your organization uses Citrix CloudPlatform to deploy and manage the cloud environment, users can use the *AutoScale* feature in CloudPlatform, in conjunction with a Citrix NetScaler appliance, to automatically scale their applications as needed. The AutoScale feature is part of the elastic load balancing feature in CloudPlatform. A CloudPlatform user can use the AutoScale feature to specify thresholds for various conditions for automatically scaling the application fleet upward and downward. CloudPlatform, in turn, configures the NetScaler appliance (by using the NetScaler NITRO API) to load balance traffic to the application virtual machines (VMs), monitor application thresholds and performance, and trigger scale-up and scale-down actions to add or remove VMs to or from the application fleet.

As the NetScaler administrator, you do not have to perform any tasks for configuring AutoScale on the NetScaler appliance. However, you might have to be aware of certain prerequisites, and you might have to troubleshoot the configuration if issues arise in the AutoScale configuration. To troubleshoot the configuration, you have to be aware of how CloudPlatform works and what configuration CloudPlatform pushes to the NetScaler appliance. You also need a working knowledge of how to troubleshoot issues on a NetScaler appliance.

For more information about AutoScale, see "AutoScale: Automatic Scaling in the Citrix CloudPlatform Environment."

Hardware Installation

Oct 25, 2013

The following sections describe the hardware installation and initial configuration for all NetScaler hardware platforms.

Introduction to the Hardware Platforms	Describes the NetScaler hardware platforms and provides detailed information about each platform and its components.
Preparing for Installation	Describes how to unpack the NetScaler appliance and prepare the site and rack for installing the appliance. Lists the cautions and warnings that you should review before you install the appliance.
Installing the Hardware	Describes the steps to install the rails, mount the hardware, connect the cables, and turn on the appliance.
Initial Configuration	Describes how to perform initial configuration of your NetScaler appliance and assign management and network IP addresses.
Lights Out Management Port of the NetScaler Appliance	Describes the different operations you can perform on your NetScaler appliance by using the Lights Out Management Port.

For information about NetScaler hardware and software compatibility and the supported upgrade and downgrade paths, see http://support.citrix.com/article/CTX113357.

Common Hardware Components

Apr 01, 2016

Each platform has front panel and back panel hardware components. The front panel has an LCD display and an RS232 serial console port. The number, type, and location of ports—copper Ethernet, copper and fiber 1G SFP, 10G SFP+, and XFP—vary by hardware platform. The back panel provides access to the fan and the field replaceable units (power supplies, CompactFlash card, and solid-state and hard-disk drives).

This document includes the follwing details:

- LCD Display and LED Status Indicators
- Ports

The LCD display on the front of every appliance displays messages about the current operating status of the appliance. These messages communicate whether your appliance has started properly and is operating normally. If the appliance is not operating normally, the LCD displays troubleshooting messages.

The LCD displays real-time statistics, diagnostic information, and active alerts. The dimensions of the LCD limit the display to two lines of 16 characters each, causing the displayed information to flow through a sequence of screens. Each screen shows information about a specific function.

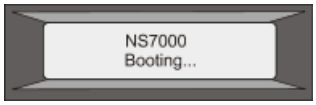
The LCD has a neon backlight. Normally, the backlight glows steadily. When there is an active alert, it blinks rapidly. If the alert information exceeds the LCD screen size, the backlight blinks at the beginning of each display screen. When the appliance shuts down, the backlight remains on for one minute and then automatically turns off.

There are nine types of display screens on the LCD display. The first two screens in the following list, the booting screen and the startup screen, appear when your appliance is starting up. The other screens, except the out-of-service screen, can appear while the appliance is operating. They show configuration information, alerts, HTTP information, network traffic information, CPU load information, and port information for your appliance.

Booting Screen.

The booting screen is displayed immediately after the appliance is turned on. The first line displays the hardware platform, as shown in the following figure.

Figure 1. LCD Booting Screen

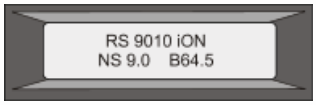


The newer MPX appliances display NSMPX followed by the platform number in the first line. For example, the MPX 7500/9500 appliances display NSMPX-7500. To view the model number, at the NetScaler command line, type show license. Scroll to the end of the command output to view the model number.

Startup Screen.

The startup screen is displayed for a few seconds after the appliance successfully begins operation. The first line displays the hardware platform, and the second line displays the software version and build number, as shown in the following figure.

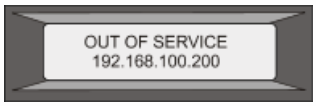
Figure 2. LCD Startup Screen



Out-of-Service Screen.

The out-of-service screen is displayed when the appliance has undergone a controlled shutdown, as shown in the following figure.

Figure 3. LCD Out-of-service Screen



Configuration Screen.

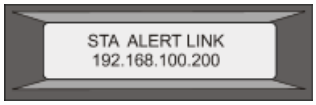
The first line displays the appliance status (STA, PRI, or SEC) and uptime. STA indicates that the appliance is in standalone mode, PRI indicates that the appliance is a primary node in a high availability (HA) pair, and SEC indicates that the appliance is a secondary node in an HA pair. Appliance uptime is displayed in HH:MM format. The second line displays the IP address of the appliance, as shown in the following figure.

Figure 4. LCD Configuration Screen

Alert Screen.

The first line displays the appliance status (STA, PRI, or SEC). STA indicates that the appliance is in standalone mode, PRI indicates that the appliance is a primary node in a high availability (HA) pair, and SEC indicates that the appliance is a secondary node in an HA pair. The second line displays the IP address of the appliance.

Figure 5. LCD Known Alert Screen



HTTP Statistics Screen.

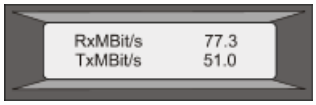
The first line displays the rate of HTTP GETS per second. The second line displays the rate of HTTP POSTS per second, as shown in the following figure.

Figure 6. LCD HTTP Statistics Screen

Network Traffic Statistics Screen.

The first line displays the rate at which data is received, in megabits per second. The second line displays the rate of data transmission, in megabits per second, as shown in the following figure.

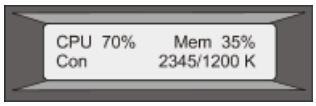
Figure 7. LCD Network Traffic Statistics Screen



CPU Load, Memory, and Connections Screen.

The first line displays CPU utilization and memory utilization as percentages. The second line displays the ratio of the

number of server connections to the number of client connections.
Note: If the number of server or client connections exceeds 99,999, the number is displayed in thousands, indicated by the letter K.
Figure 8. LCD CPU Load, Memory, and Connections Screen



Port Information Screen.

The S row displays port speed, flow control, and duplex information. The R row displays megabits received per second on the interface. The first port in each row is the management port.
Figure 9. Port Information for an 8-port Appliance

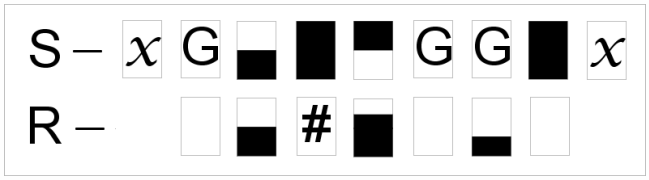
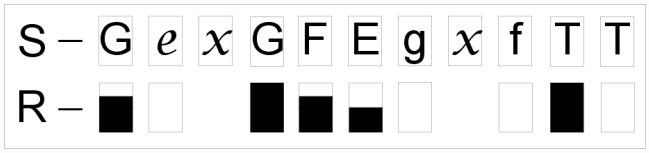


Figure 10. Port Information for a 10-port Appliance



The following table defines the various abbreviations and symbols that appear in the S row of the port information screen.

Table 1. Port Abbreviations and Symbols for S Row

S row abbreviation/symbol	Indicates
E	A rate of 10 megabits per second, full duplex mode, and flow control OFF.
F	A rate of 100 megabits per second, full duplex mode, and flow control OFF.
G	A rate of 1 gigabit per second, full duplex mode, and flow control OFF.
T	A rate of 10 gigabits per second, full duplex mode, and flow control OFF.
	A disconnected port.

S row abbreviation/symbol	Indicates
x	Note: The R row does not display an abbreviation or symbol for a disconnected port.
	Receive flow control regardless of speed or duplex mode.
	Transmit flow control regardless of speed or duplex mode.
	Receive and transmit flow control regardless of speed or duplex mode.
e	A rate of 10 megabits per second, half duplex mode, and flow control OFF.
f	A rate of 100 megabits per second, half duplex mode, and flow control OFF.
g	A rate of 1 gigabit per second, half duplex mode, and flow control OFF.

The following table defines the various abbreviations and symbols that appear in the R row of the port information screen.

Table 2. Port Abbreviations and Symbols for R Row

R row abbreviation/symbol	Indicates
#	The port is disabled.
	Receive speed is about 10% of line speed.
	Receive speed is about 50% of line speed.

R row abbreviation/symbol	Indicates
	Receive speed is about 75% of line speed.
	Receive speed is about 100% of line speed.

On the appliance's back panel, system status LEDs indicate the overall status of the appliance. The following table describes the indicators of the system status LED.

Note: System status LEDs are available on only the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 appliances.

LED Color	LED Indicates
OFF	No power
Green	Appliance is receiving power
Red	Appliance has detected an error

On the appliance's back panel, power status LEDs indicate the status of each power supply. The following table describes the indicators of the power status LED.

LED Color	LED Indicates
OFF	No power
Green	Appliance is receiving power
Red	Power supply has detected an error

The port LEDs show whether a link is established and traffic is flowing through the port. The following table describes the LED indicators for each port. There are two LED indicators for each port type.

Note: This section applies to the MPX 5500, MPX 5550/5650, MPX 7500/9500, MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 17500/19500/21500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14020/14030/14040/14060/14080/14100, MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, MPX 25100T/25160T appliances.

Table 3. LED port-status indicators

Port Type	LED Location	LED Function	LED Color	LED Indicates
10G SFP+ (10 Gbps)	Top	Speed	Off	No connection.
			Solid blue	Traffic rate of 10 gigabits per second.
	Bottom	Link/ Activity	Off	No link.
			Solid green	Link is established but no traffic is passing through the port.
			Blinking green	Traffic is passing through the port.
1G SFP (1 Gbps)	Left	Link/ Activity	Off	No link.
			Solid green	Link is established but no traffic is passing through the port.
			Blinking green	Traffic is passing through the port.
	Right	Speed	Off	No connection.
			Yellow	Traffic rate of 1 gigabit per second.
Ethernet (RJ45)	Left	Speed	Off	No connection, or a traffic rate of 10 megabits per second (Mbps).
			Green	Traffic rate of 100 Mbps.
			Yellow	Traffic rate of 1 gigabit per second.
	Right	Link/ Activity	Off	No link.
			Solid green	Link is established but no traffic is passing through the port.
			Blinking green	Traffic is passing through the port.
Management	Left	Speed	Off	No connection, or a traffic rate of 10 megabits per

Port Type	LED Location	LED Function	LED Color	LED Indicates
(RJ45)			Green	second (Mbps). Traffic rate of 100 Mbps.
			Amber	Traffic rate of 1 gigabit per second.
	Right	Link/ Activity	Off	No link.
			Solid yellow	Link is established but no traffic is passing through the port.
			Blinking yellow	Traffic is passing through the port.

On each power supply, a bicolor LED indicator shows the condition of the power supply. The LEDs of the AC power supplies for MPX 15000 and 17000 appliances are different from the LEDs of the other appliances.

Table 4. LED Power Supply Indicators

Power Supply Type	LED Color	LED Indicates
AC	OFF	No power to any power supply.
	Flashing RED	No power to this power supply.
	Flashing GREEN	Power supply is in standby mode.
	GREEN	Power supply is functional.
	RED	Power supply failure.
DC	OFF	No power to any power supply.
	Flashing RED	No power to this power supply.
	Flashing BLUE	Power supply is in standby mode.
	BLUE	Power supply is functional.
	RED	Power supply failure.

Power Supply Type	LED Color	LED Indicates
MPX 15000 and 17000	OFF	Power supply is not plugged in to a power source. If the LED is off when the power supply is plugged in, the power supply has a malfunction.
	AMBER	Power supply has been plugged in for less than a few seconds. If the LED does not turn GREEN, the power supply has a malfunction.
	GREEN	Power supply is functioning properly.
	BLINKING	Power supply has a malfunction

Ports are used to connect the appliance to external devices. NetScaler appliances support RS232 serial ports, 10/100/1000Base-T copper Ethernet ports, 1-gigabit copper and fiber 1G SFP ports, and 10-gigabit fiber SFP+ ports. All NetScaler appliances have a combination of some or all of these ports. For details on the type and number of ports available on your appliance, see the section describing that platform.

RS232 Serial Port

The RS232 serial console port provides a connection between the appliance and a computer, allowing direct access to the appliance for initial configuration or troubleshooting.

All hardware platforms ship with an appropriate serial cable used to connect your computer to the appliance. For instructions on connecting your computer to the appliance, see "Installing the Hardware."

Copper Ethernet Ports

The copper Ethernet ports installed on many models of the appliance are standard RJ45 ports.

There are two types of copper Ethernet ports that may be installed on your appliance:

10/100BASE-T port

The 10/100BASE-T port has a maximum transmission speed of 100 megabits per second (Mbps). Most platforms have at least one 10/100BASE-T port.

10/100/1000BASE-T port

The 10/100/1000BASE-T port has a maximum transmission speed of 1 gigabit per second, ten times faster than the other type of copper Ethernet port. Most platforms have at least one 10/100/1000Base-T port.

To connect any of these ports to your network, you plug one end of a standard Ethernet cable into the port and plug the other end into the appropriate network connector.

Management Ports

Management ports are standard copper Ethernet ports (RJ45), which are used for direct access to the appliance for system administration functions.

1G SFP, 10G SFP+, and XFP Ports

A 1G SFP port can operate at a speed of 1 Gbps. It accepts either a copper 1G SFP transceiver, for operation as a copper Ethernet port, or a fiber 1G SFP transceiver for operation as a fiber optic port.

The 10G SFP+ and XFP ports are high-speed ports that can operate at speeds of up to 10 Gbps. You need a fiber optic cable to connect to a 10G SFP+ or XFP port. If the other end of the fiber optic cable is attached to a 1G SFP port, the 10G SFP+ port automatically negotiates to match the speed of the 1G SFP port.

Ports Compatibility

The 10G slot supports **copper** 1G transceivers, which can operate at up to 1 Gbps in a 10 Gbps slot.

Note: You cannot insert a copper 1G transceiver into a 10G slot in MPX 14060/14080-40G, and MPX 25100/25160-40G appliances.

Note: You cannot insert a fiber 1G transceiver into a 10G slot.
Note: You cannot insert a 10G transceiver into a 1G slot.
The following tables list the maximum distance specifications for NetScaler pluggable media (1G SFP, 10G SFP+, and XFP transceivers).
Note: The tables are categorized by 1G pluggable media and 10G pluggable media.
The 10G SFP+ modules are dual-speed capable and support both 1G and 10G, depending on the peer switch that the model connects to. These are listed in both tables.

Both tables have the following columns:

- SKU: Citrix maintains multiple SKUs for the same part.
- Description: The price list description of the part.
- Transmit Wavelength: The nominal transmit wavelength.
- Cable/Fiber Type: Fiber characteristics affect the maximum transmit distance achievable. This is especially true with 10G on multi-mode fiber (MMF), where various dispersion components become dominant. For more information, see http://www.thefoa.org/tech/ref/basic/fiber.html.
- Typical Reach: Maximum transmit distance.
- Products: Some chassis are available with different media options. Use the appropriate data sheet to confirm that your particular chassis type supports the media.

1G Pluggable Media

The following table lists the maximum distance specifications for 1G transceivers.

Table 5. Copper 1G SFP Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Cable Type	Typical Reach (m)	Products
EW3A0000235, EW3B0000235, EW3C0000235, EW3D0000235, EW3E0000235, EW3F0000235, EW3P0000143, EW3X0000235, EW3Z0000087	Citrix NetScaler 1G SFP Ethernet Copper (100m) - **4 Pack**	n/a	Category 5 (Cat-5) Copper Cable	100 m	MPX 7500/9500, MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, 9010 FIPS, MPX 22040/22060/22080/22100/22120, MPX 24100/24150

Table 6. Short Reach Fiber 1G SFP Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000234, EW3B0000234, EW3C0000234, EW3D0000234, EW3E0000234, EW3F0000234, EW3P0000142, EW3X0000234, EW3Z0000086	Citrix NetScaler 1G SFP Ethernet SX (300m) - **4 Pack**	850nm (nominal)	50/125um MMF, 2000MHz-km (OM3)	550 m	MPX 7500/9500, MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, 9010 FIPS, MPX 22040/22060/22080/22100/22120, MPX 24100/24150
			50/125um MMF, 500MHz-km (OM2)	550 m	
			50/125um MMF, 400MHz-km	550 m	
			62.5/125um MMF,	300 m	

			200MHz-km (OM1)		
			62.5/125um MMF, 160MHz-km	300 m	

Table 7. Short Reach Fiber 1G SFP Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000710, EW3B0000710, EW3C0000710, EW3D0000710, EW3E0000710, EW3F0000710, EW3P0000557, EW3X0000710, EW3Z0000585	Citrix NetScaler 1G SFP Ethernet Short Range (300m) - **Single**	850nm (nominal)	50/125um MMF, 2000MHz-km (OM3)	550 m	MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 17500/19500/21500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542,, MPX 22040/22060/22080/22100/22120, MPX 24100/24150
			50/125um MMF, 500MHz-km (OM2)	550 m	
			50/125um MMF, 400MHz-km	550 m	
			62.5/125um MMF, 200MHz-km (OM1)	275 m	
			62.5/125um MMF, 160MHz-km	220 m	

Table 8. Long Reach Fiber 1G SFP Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000712, EW3B0000712, EW3C0000712, EW3D0000712, EW3E0000712, EW3F0000712, EW3P0000559, EW3X0000712, EW3Z0000587	Citrix NetScaler 1G SFP Ethernet LX - **Single**	1310nm (nominal)	9/125um SMF	10 km	MPX 7500/9500, MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, 9010 FIPS, MPX 22040/22060/22080/22100/22120, MPX 24100/24150

Table 9. Long Reach Fiber 1G SFP Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000711, EW3B0000711, EW3C0000711, EW3D0000711,	Citrix NetScaler 1G	1310nm (nominal)	9/125um SMF	10 km	MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX

EW3E0000711, EW3F0000711, EW3P0000558, EW3X0000711, EW3Z0000586	SFP Ethernet Long Range (10km) - **Single**				17500/19500/21500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542,, MPX 22040/22060/22080/22100/22120, MPX 24100/24150

10 GE Pluggable Media

The following table lists the maximum distance specifications for 10G transceivers.

Table 10. Short Reach Fiber 10G SFP+ Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000710, EW3B0000710, EW3C0000710, EW3D0000710, EW3E0000710, EW3F0000710, EW3P0000557, EW3X0000710, EW3Z0000585	Citrix NetScaler 10G SFP+ Ethernet Short Range (300m) - **Single**	850nm (nominal)	50/125um MMF, 2000MHz-km (OM3)	300 m	MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 17500/19500/21500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14020/14030/14040/14060/14080/14100, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, MPX 25100T/25160T,MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G
			50/125um MMF, 500MHz-km (OM2)	82 m	
			50/125um MMF, 400MHz-km	66 m	
			62.5/125um MMF, 200MHz-km (OM1)	33 m	
			62.5/125um MMF, 160MHz-km	26 m	

Table 11. Short Reach XFP (10G) Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000713, EW3B0000713, EW3C0000713, EW3D0000713, EW3E0000713, EW3F0000713, EW3P0000560, EW3X0000713, EW3Z0000588	Citrix NetScaler XFP Short Range 10 Gigabit Ethernet(300m) - **Single**	850nm (nominal)	50/125um MMF, 2000MHz-km (OM3)	300 m	MPX 15000/17000
			50/125um MMF, 500MHz-km (OM2)	82 m	
			50/125um MMF,	66 m	

			400MHz-km		
			62.5/125um MMF, 200MHz-km (OM1)	33 m	
			62.5/125um MMF, 160MHz-km	26 m	

Table 12. Long Reach Fiber 10G SFP+ Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000711, EW3B0000711, EW3C0000711, EW3D0000711, EW3E0000711, EW3F0000711, EW3P0000558, EW3X0000711, EW3Z0000586	Citrix NetScaler 10G SFP+ Ethernet Long Range (10km) - **Single**	1310nm (nominal)	9/125um SMF	10 km	MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 17500/19500/21500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14020/14030/14040/14060/14080/14100, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, MPX 25100T/25160T,MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G

Table 13. Long Reach Fiber XFP (10G) Distance Specifications

SKU	Description	Transmitter Wavelength (nm)	Fiber Type	Typical Reach (m)	Products
EW3A0000714, EW3B0000714, EW3C0000714, EW3D0000714, EW3E0000714, EW3F0000714, EW3P0000561, EW3X0000714, EW3Z0000589	Citrix NetScaler XFP Long Range 10 Gigabit Ethernet(10 km) - **Single**	1310nm (nominal)	9/125um SMF	10 km	MPX 15000/17000

Table 14. Cisco 40G QSFP+ Cable Specifications

Cisco Part Number	Description	Products
L45593-D178-C30	40GBASE-CR4 QSFP+ to four 10GBASE-CU SFP+ direct attach breakout cable assembly, 3 meter passive	MPX 11500/13500/14500/16500/18500/20500, MPX 17500/19500/21500, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, MPX 25100T/25160T,MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G
Note: Support for this cable is available in release 10.1 build 122.17 and later. Note: **To obtain these cables, contact Cisco partner representatives.**		

Field Replaceable Units

Apr 04, 2016

Citrix NetScaler field replaceable units (FRU) are NetScaler components that can be quickly and easily removed from the appliance and replaced by the user or a technician at the user's site. The FRUs in a NetScaler appliance can include a CompactFlash card, DC or AC power supplies, and solid-state or hard-disk drives, and a direct attach cable (DAC).

Note: The solid-state or hard-disk drive stores your configuration information, which has to be restored from a backup after replacing the unit.

This document includes the following details:

For appliances containing two power supplies, the second power supply acts as a backup. The MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 appliances can accommodate four power supplies, and require two power supplies for proper operation. The third and fourth power supplies act as backup.

The appliance ships with a standard power cord that plugs into the appliance's power supply and an NEMA 5-15 plug on the other end for connecting to the power outlet on the rack or in the wall.

For power-supply specifications, see "Hardware Platforms," which describes the various platforms and includes a table summarizing the hardware specifications.

Note: If you suspect that a power-supply fan is not working, please see the description of your platform. On some platforms, what appears to be the fan does not turn, and the actual fan turns only when necessary.

On each power supply, a bicolor LED indicator shows the condition of the power supply.

Electrical Safety Precautions for Power Supply Replacement

- Make sure that the appliance has a direct physical connection to earth ground during normal use. When installing or repairing an appliance, always connect the ground circuit first and disconnect it last.
- Always unplug any appliance before performing repairs or upgrades.
- Never touch a power supply when the power cord is plugged in. As long as the power cord is plugged in, line voltages are present in the power supply even if the power switch is turned off.

Replacing an AC Power Supply

Citrix NetScaler MPX platforms can accommodate two power supplies, except the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 platforms which can accommodate four power supplies. All NetScaler appliances function properly with a single power supply, except the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 platforms which need two power supplies for proper operation. The other power supplies serves as a backup. All power supplies must be of the same type (AC or DC).

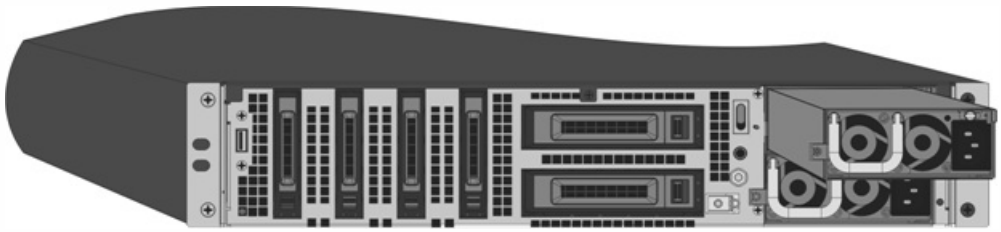
Note: If the appliance has only one power supply, you have to shut down the appliance before replacing the power supply.

If the appliance has two power supplies, you can replace one power supply without shutting down the appliance, provided the other power supply is working, and if the appliance has four power supplies, you can replace one or two power supplies without shutting down the appliance, provided the other two power supplies are working.

To install or replace an AC power supply on a Citrix NetScaler appliance

1. Align the semicircular handle perpendicular to the power supply. Loosen the thumbscrew and press the lever toward the handle and pull out the existing power supply, as shown in the following figure.
   Note: The illustration in the following figures might not represent the actual NetScaler appliance.
   Figure 1. Removing the Existing AC Power Supply

2. Carefully remove the new power supply from its box.
3. On the back of the appliance, align the power supply with the power supply slot.
4. Insert the power supply into the slot and press against the semicircular handle until you hear the power supply snap into place.
   Figure 2. Inserting the Replacement AC Power Supply



5. Connect the power supply to a power source. If connecting all power supplies, plug separate power cords into the power supplies and connect them to separate wall sockets.

Note: NetScaler appliances emit a high-pitched alert if one power supply fails or if you connect only one power cable to an appliance in which two power supplies are installed. To silence the alarm, press the small red button on the back panel of the appliance. The disable alarm button is functional only when the appliance has two power supplies.

Replacing a DC Power Supply

Citrix NetScaler MPX platforms can accommodate two power supplies, except the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 platforms which can accommodate four power supplies. All NetScaler appliances function properly with a single power supply, except the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 platforms which need two power supplies for proper operation. The other power supplies serves as a backup. All power supplies must be of the same type (AC or DC).

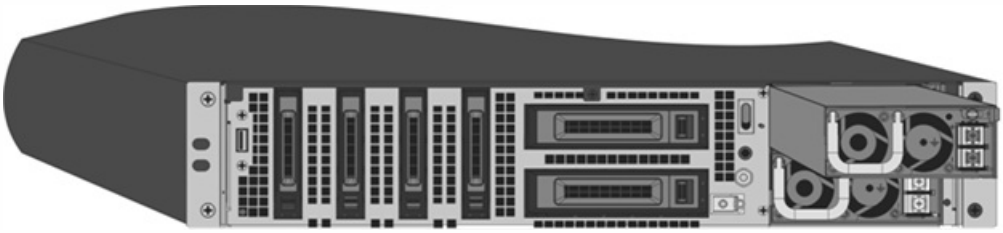
Note: If the appliance has only one power supply, you have to shut down the appliance before replacing the power supply. If the appliance has two power supplies, you can replace one power supply without shutting down the appliance, provided

the other power supply is working, and if the appliance has four power supplies, you can replace one or two power supplies without shutting down the appliance, provided the other two power supplies are working.

To install or replace a DC power supply on a Citrix NetScaler appliance

1. Loosen the thumbscrew and press the lever towards the handle and pull out the existing power supply, as shown in the following figure.
   Note: The illustration in the following figures might not represent the actual NetScaler appliance.
   Figure 3. Removing the Existing DC Power Supply

2. Carefully remove the new power supply from its box.
3. On the back of the appliance, align the power supply with the power supply slot.
4. Insert the power supply into the slot while pressing the lever towards the handle. Apply firm pressure to insert the power supply firmly into the slot.
   Figure 4. Inserting the Replacement DC Power Supply



5. When the power supply is completely inserted into its slot, release the lever.
6. Connect the power supply to a power source. If connecting all power supplies, plug separate power cords into the power supplies and connect them to separate wall sockets.

Note: NetScaler appliances emit a high-pitched alert if one power supply fails or if you connect only one power cable to an appliance in which two power supplies are installed. To silence the alarm, press the small red button on the back panel of the appliance. The disable alarm button is functional only when the appliance has two power supplies.

The NetScaler software is stored on either the solid-state drive or the CompactFlash card. The following MPX platforms store the NetScaler software on the CompactFlash card:

- Citrix NetScaler MPX 5500
- Citrix NetScaler MPX 7500 and MPX 9500
- Citrix NetScaler MPX 9700, MPX 10500, MPX 12500, and MPX 15500
- Citrix NetScaler MPX 15000
- Citrix NetScaler MPX 17000

Note: The CompactFlash card is mounted as /flash on the above platforms.
The CompactFlash card specifications vary by NetScaler hardware platform. A CompactFlash card from one platform does not necessarily work on a different platform.

Replacing a CompactFlash Card

Note: These instructions apply to the Citrix® NetScaler® MPX 5500, MPX 7500/9500, MPX 9700/10500/12500/15500, MPX 15000, and MPX 17000 appliances only.
Replacement CompactFlash cards contain a preinstalled version of the NetScaler software and a generic configuration file (ns.conf), but they do not contain SSL-related certificates and keys, or custom boot settings. Configuration files and customized settings must be restored from a backup storage location at the customer site, if available. The files to be restored might include:

- /flash/nsconfig/ns.conf: The current configuration file.
- /flash/nsconfig/ZebOS.conf: The ZebOS configuration file.
- /flash/nsconfig/license: The licenses for the NetScaler features.
- /flash/nsconfig/ssl: The SSL certificates and keys required for encrypting data to clients or to backend servers.
- /nsconfig/rc.netscaler: Customer-specific boot operations (optional).

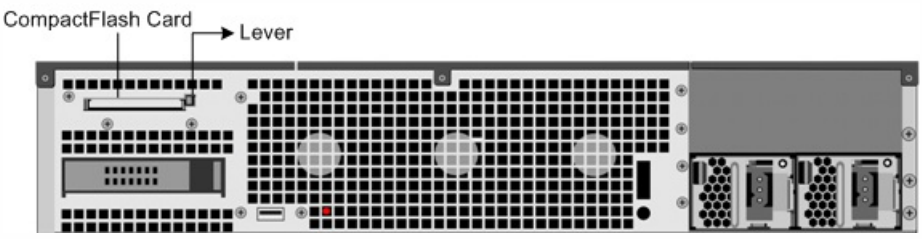
Note: Verify that the card you receive is the correct type for your NetScaler appliance.

To replace a CompactFlash card

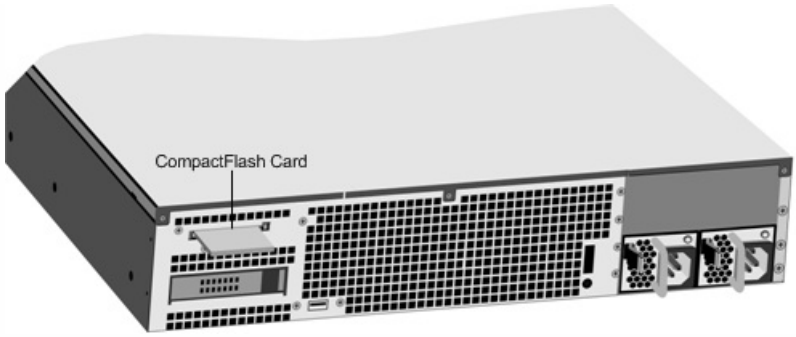
1. At the NetScaler command prompt, exit to the shell prompt. Type:
   shell

2. Shut down the NetScaler appliance by typing one of the following commands at the shell prompt.
   - On an MPX appliance, type:
     shutdown -p now
   - On a non-MPX appliance, type:
     shutdown

3. Locate the CompactFlash slot on the back panel of the appliance.
4. Disengage the CompactFlash by pushing the lever to the right of the CompactFlash slot. If necessary, use a pen or small screwdriver to push the lever in fully. Pull the existing flash card out of the slot.
   Note: The illustration in the following figures might not represent the actual NetScaler appliance.
   Figure 5. Removing the Existing CompactFlash Card



5. Insert the new flash card received from Citrix.
   Important: When you insert the card, make sure that the arrow on top of the card is pointing toward the CompactFlash slot. Position the connector grid on the edge of the CompactFlash card to meet the matching connector pins inside the CompactFlash slot.
   Figure 6. Inserting the Replacement CompactFlash Card



6. Turn on the NetScaler appliance.
   When the appliance starts, it no longer has the previous working configuration. Therefore, the appliance is reachable only through the default IP address of 192.168.100.1/16, or through the console port.
7. Perform the initial configuration of the appliance, as described in "Initial Configuration." Log on to the default IP address by using a web browser, or connect to the serial console by using a console cable, to perform the initial configuration.
8. Upload a platform license and any optional feature licenses, including universal licenses, to the NetScaler appliance. For more information, see the licensing chapter of the "Getting Started with Citrix NetScaler."
9. Once the correct NetScaler software version is loaded, you can restore the working configuration. Copy a previous version of the ns.conf file to the /nsconfig directory by using an SCP utility or by pasting the previous configuration into the /nsconfig/ns.conf file from the NetScaler command prompt. To load the new ns.conf file, restart the NetScaler appliance by entering the reboot command at the NetScaler command prompt.

A solid-state drive (SSD) is a high-performance device that stores data in solid-state flash memory. The MPX solid-state drives contain the boot loader configuration file, configuration file (ns.conf), licenses, and for some models, the NetScaler software and the user data.
The NetScaler software is stored on either the SSD or the CompactFlash card. The following MPX platforms store the NetScaler software on the SSD. The SSD is mounted as /flash.

- Citrix NetScaler MPX 5550 and MPX 5650
- Citrix NetScaler MPX 8005, MPX 8015, MPX 8200, MPX 8400, MPX 8600, and MPX 8800
- Citrix NetScaler MPX 11500, MPX 13500, MPX 14500, MPX 16500, MPX 18500, and MPX 20500
- Citrix NetScaler MPX 11515, MPX 11520, MPX 11530, MPX 11540, and MPX 11542
- Citrix NetScaler MPX 14020/14030/14040/14060/14080/14100
- Citrix NetScaler MPX 17500, MPX 19500, and MPX 21500
- Citrix NetScaler MPX 17550, MPX 19550, MPX 20550, and MPX 21550
- Citrix NetScaler MPX 22040, MPX 22060, MPX 22080, MPX 22100, and MPX 22120
- Citrix NetScaler MPX 24100 and MPX 24150
- Citrix NetScaler MPX 25100T and MPX 25160T
- MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G
- MPX 25100 40G, MPX 25160 40G

Note: On the MPX 5550/5650 and MPX 8005/8015/8200/8400/8600/8800 appliances, both /flash and /var are mounted from different partitions of the same SSD drive.

Replacing a Solid-State Drive

Note: These instructions apply to the Citrix NetScaler MPX 5550/5650, MPX 8005/8015/8200/8400/8600/8800, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14020/14030/14040/14060/14080/14100, MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G, MPX 17500/19500/21500, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, and MPX 25100T/25160T appliances.
Replacement solid-state drives (SSDs) contain a pre-installed version of the NetScaler software and a generic configuration file (ns.conf), but they do not contain SSL-related certificates and keys, or custom boot settings. Configuration files and customized settings must be restored to a replacement drive from a backup storage location at the customer site, if available. The files to be restored might include:

- /flash/nsconfig/ns.conf: The current configuration file.
- /flash/nsconfig/ZebOS.conf: The ZebOS configuration file.
- /flash/nsconfig/license: The licenses for the NetScaler features.
- /flash/nsconfig/ssl: The SSL certificates and keys required for encrypting data to clients or to backend servers.

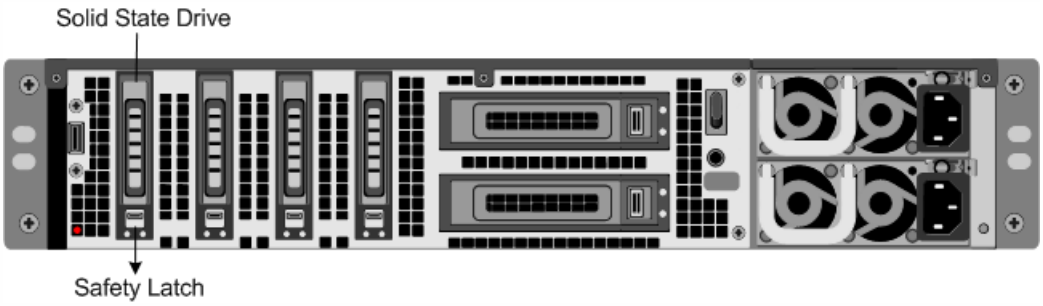
- /nsconfig/rc.netscaler: Customer-specific boot operations (optional).

To replace a solid-state drive

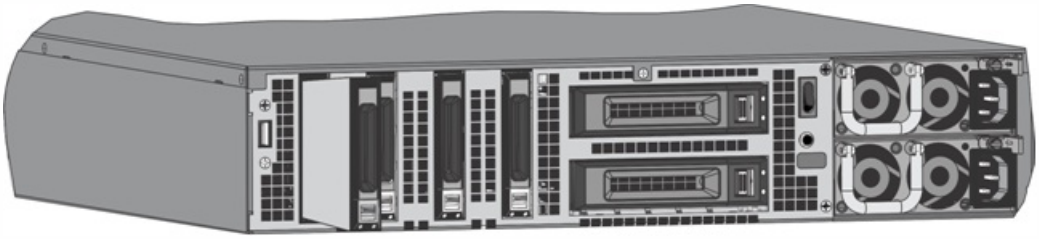
1. At the NetScaler command prompt, exit to the shell prompt. Type:
   shell

2. Shut down the NetScaler appliance by typing the following command at the shell prompt:
   shutdown –p now

3. Locate the SSD on the back panel of the appliance. Push the safety latch of the drive cover to the right or down, depending on the platform, while pulling out on the drive handle to disengage. Pull out the faulty drive.
   Note: The illustration in the following figures might not represent the actual NetScaler appliance.
   Figure 7. Removing the Existing Solid-State Drive



4. Verify that the replacement SSD is the correct type for the platform.
5. Pick up the new SSD, open the drive handle fully to the left or up, and insert the drive into the slot as far as possible. To seat the drive, close the handle flush with the rear of the appliance so that the drive locks securely into the slot.
   Important: When you insert the drive, make sure that the Citrix product label is at the top if the drive is inserted horizontally or at the right if the drive is inserted vertically.
   Figure 8. Inserting the Replacement Solid-State Drive



6. Turn on the NetScaler appliance. When the appliance starts, it no longer has the previous working configuration. Therefore, the appliance is reachable only through the default IP address of 192.168.100.1/16, or through the console port.
7. Perform the initial configuration of the appliance, as described in "Initial Configuration." Log on to the default IP address by using a web browser, or connect to the serial console by using a console cable, to perform the initial configuration.
8. Upload a platform license and any optional feature licenses, including universal licenses, to the NetScaler appliance. For more information, see the licensing chapter of the "Getting Started with Citrix NetScaler."
9. Once the correct NetScaler software version is loaded, you can restore the working configuration. Copy a previous version of the ns.conf file to the /nsconfig directory by using an SCP utility or by pasting the previous configuration into the /nsconfig/ns.conf file from the NetScaler command prompt. To load the new ns.conf file, you must restart the NetScaler appliance by entering the reboot command at the NetScaler command prompt.

A hard disk drive (HDD) stores logs and other data files. Files stored on the HDD include the newnslog files, dmesg and messages files, and any core/crash files. The HDD comes in various capacities, depending on the Citrix NetScaler platform. Hard drives are used for storing files required at runtime. An HDD is mounted as /var.

The following MPX platforms support HDD:

- Citrix NetScaler MPX 9700, MPX 10500, MPX 12500, and MPX 15500
- Citrix NetScaler MPX 11500, MPX 13500, MPX 14500, MPX 16500, MPX 18500, and MPX 20500

- Citrix NetScaler MPX 11515, MPX 11520, MPX 11530, MPX 11540, and MPX 11542
- Citrix NetScaler MPX 15000
- Citrix NetScaler MPX 17000
- Citrix NetScaler MPX 17500, MPX 19500, and MPX 21500
- Citrix NetScaler MPX 17550, MPX 19550, MPX 20550, and MPX 21550
- Citrix NetScaler MPX 22040, MPX 22060, MPX 22080, MPX 22100, and MPX 22120
- Citrix NetScaler MPX 24100 and MPX 24150

Replacing a Hard Disk Drive

A hard disk drive (HDD) stores log files and other user files. Collection of new log files begins upon boot-up with the new HDD. Product documentation can be downloaded from "MyCitrix.com" and reinstalled to the /var/netscaler/doc location.

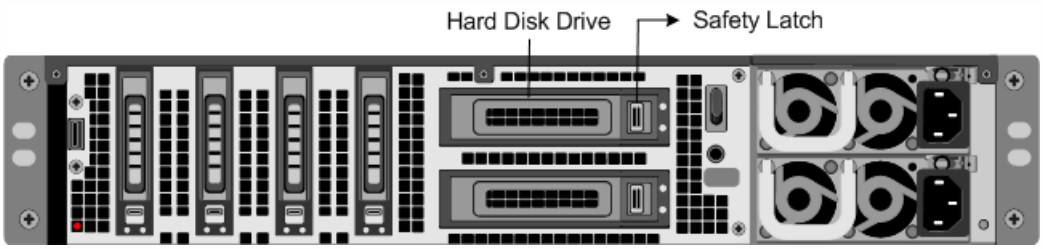
To install a hard disk drive

1. At the NetScaler command prompt, exit to the shell prompt. Type:
   shell

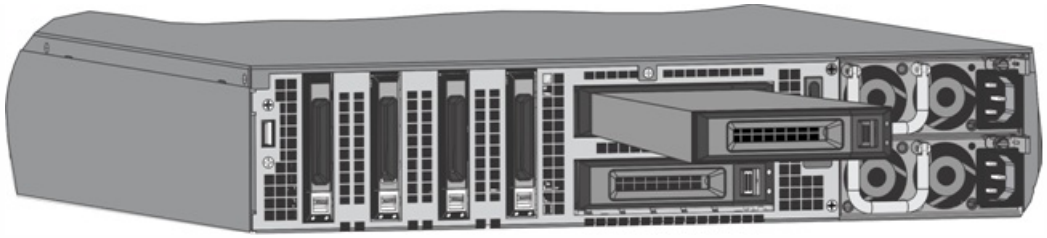
2. Shut down the NetScaler appliance by typing one of the following commands at the shell prompt.
   - On an MPX appliance, type:
     shutdown –p now

   - On a non-MPX appliance, type:
     shutdown

3. Locate the hard disk drive on the back panel of the appliance.
4. Verify that the replacement hard disk drive is the correct type for the NetScaler platform.
5. Disengage the hard disk drive by pushing the safety latch of the drive cover to the right or down, depending on the platform, while pulling out on the drive handle to disengage. Pull out the faulty drive.
   Note: The illustration in the following figures might not represent the actual NetScaler appliance.
   Figure 9. Removing the Existing Hard Disk Drive



6. Pick up the new disk drive, open the drive handle fully to the left, and insert the new drive into the slot as far as possible. To seat the drive, close the handle flush with the rear of the appliance so that the hard drive locks securely into the slot.
   Important: When you insert the drive, make sure that the Citrix product label is at the top.
   Figure 10. Inserting the Replacement Hard Disk Drive



7. Turn on the NetScaler appliance. The appliance starts the NetScaler software and reads the configuration file from the CompactFlash card.

A direct attach cable (DAC) assembly is a high performance integrated duplex data link for bi-directional communication. The cable is compliant with the IPF MSA (SFF-8432) for mechanical form factor and SFP+ MSA for direct attach cables. The cable, which can be up to 5 meters long, is data-rate agnostic. Supporting speeds in excess of 10 Gbps, it is a cost-effective alternative to optical links (SFP+ transceivers and fiber optic cables.) The transceiver with DAC is hot-swappable. You can insert and remove the transceiver with the attached cable without shutting down the appliance. The Citrix NetScaler appliance supports only passive DAC.

Important:

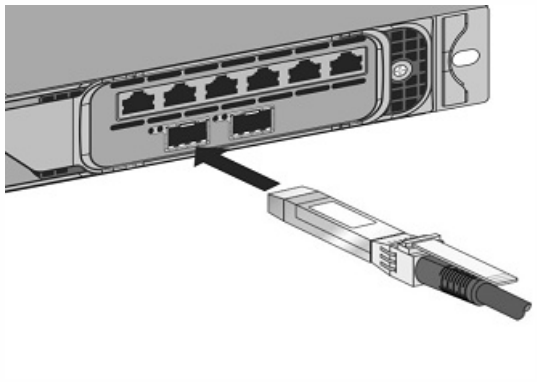
- DAC is supported only on 10G ports. Do not insert a DAC into a 1G port.
- Do not attempt to unplug the integrated copper cable from the transceiver and insert a fiber cable into the transceiver.

Installing a Direct Attach Cable

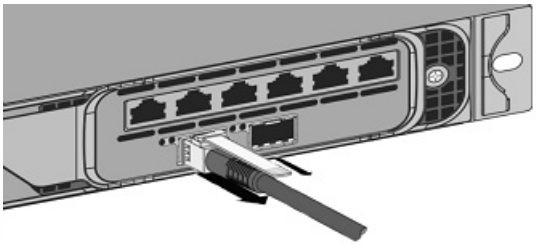
Note: The illustrations in the following figures are only for reference and might not represent the actual NetScaler appliance.

To install or remove a direct attach cable

1. To install the DAC, slide it into the 10G port on the appliance, as shown in the following figure. You will hear a click when the DAC properly fits into the port.
   Figure 11. Inserting a DAC into the 10G port



2. To remove the DAC, pull the tab on the top of the DAC, and then pull the DAC out of the port, as shown in the following figure.
   Figure 12. Removing a DAC from the 10G port



Hardware Platforms

Oct 07, 2013

The various NetScaler hardware platforms offer a wide range of features, communication ports, and processing capacities. All the MPX platforms have multicore processors.

The NetScaler hardware platforms range from the single processor MPX 5500 platform to the high-capacity, MPX 22040/22060/22080/22100/22120 hardware platform. The various NetScaler hardware platforms are similar in that they use the same types of components, but different models provide different hardware capabilities. All NetScaler hardware platforms support the NetScaler software.

Some of the hardware platforms are available as dedicated application firewall appliances or secure application access appliances.

For information on the software releases supported on the NetScaler hardware platforms, see Supported Releases on NetScaler Hardware.

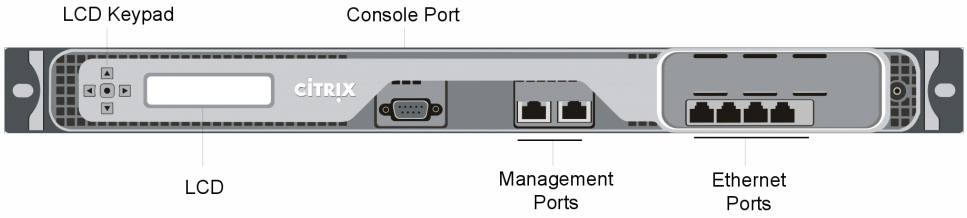
Citrix NetScaler MPX 5500

Sep 04, 2013

The Citrix NetScaler MPX 5500 is a 1U appliance, with 1 dual-core processor, and 4 gigabytes (GB) of memory.

The following figure shows the front panel of the MPX 5500.

Figure 1. Citrix NetScaler MPX 5500, front panel



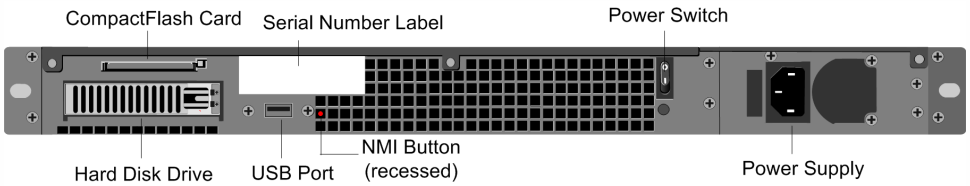
The MPX 5500 has the following ports:

- RS232 serial console port.
- Two 10/100/1000Base-T copper Ethernet management ports, numbered 0/1 and 0/2 from left to right. You can use these ports to connect directly to the appliance for system administration functions.
- Four 10/100/1000Base-T copper Ethernet ports numbered 1/1, 1/2, 1/3, and 1/4 from left to right.

Note: The network port numbers on all appliances consist of two numbers separated by a forward slash. The first number is the port adapter slot number. The second number is the interface port number. Ports on appliances are numbered sequentially starting with 1.

The following figure shows the back panel of the MPX 5500.

Figure 2. Citrix NetScaler MPX 5500, back panel



The following components are visible on the back panel of the MPX 5500:

- Four GB removable CompactFlash card that is used to store the NetScaler software.
- Power switch, which turns off power to the MPX 5500, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Removable hard-disk drive (HDD) that is used to store user data. Appliances shipped before February, 2012 store user data on a HDD. In appliances shipped after February, 2012, a solid-state drive replaces the HDD. Both types of drive have the same functionality and support the same software releases.
- USB port (reserved for a future release).
- Non-maskable interrupt (NMI) Button that is used at the request of Technical Support and produces a core dump on the NetScaler. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Power supply rated at 300 watts, 110-220 volts. The power-supply fan is designed to turn on only when the internal temperature of the power supply reaches a certain value. You cannot see the fan turning on the back panel. What you can see is the fixed part of the fan that holds the spinning motor.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

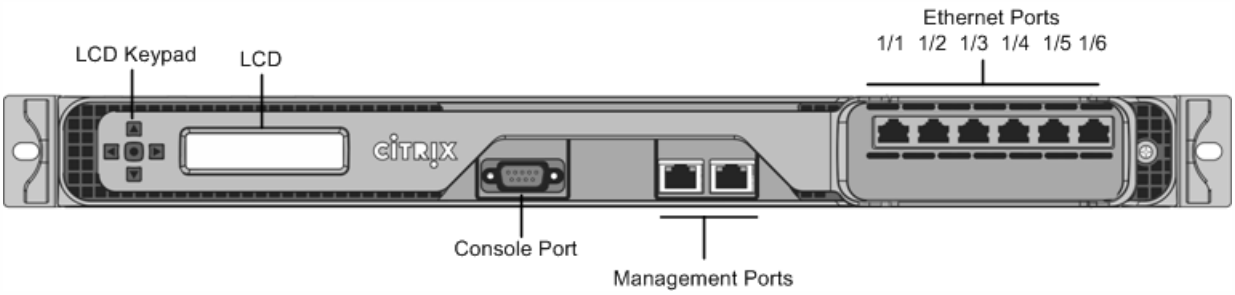
Citrix NetScaler MPX 5550 and MPX 5650

Sep 04, 2013

The Citrix NetScaler models MPX 5550 and MPX 5650 are 1U appliances. Each model has one quad-core processor and 8 gigabytes (GB) of memory.

The following figure shows the front panel of the MPX 5550/5650 appliance.

Figure 1. Citrix NetScaler MPX 5550/5650, front panel

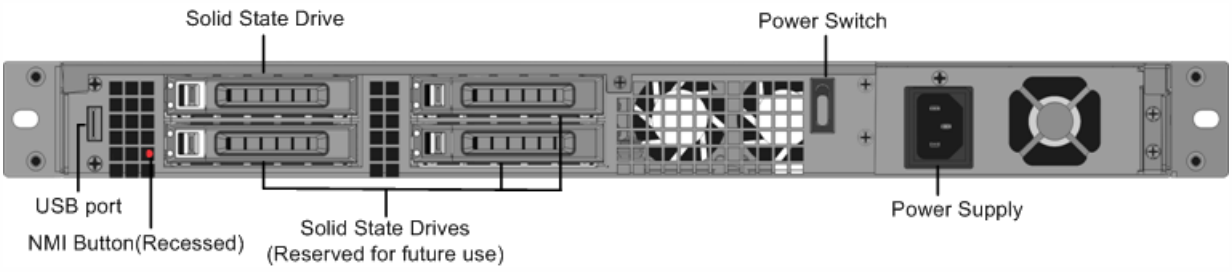


Depending on the model, the appliance has the following ports:

- RS232 serial console port.
- Two 10/100/1000Base-T copper Ethernet management ports (RJ45), numbered 0/1 and 0/2 from left to right. The management port is used to connect directly to the appliance for system administration functions.
- Six 10/100/1000Base-T copper Ethernet ports numbered 1/1, 1/2, 1/3, 1/4, 1/5, and 1/6 from left to right.

The following figure shows the back panel of the MPX 5550/5650 appliance.

Figure 2. Citrix NetScaler MPX 5550/5650 appliance, back panel



The following components are visible on the back panel of the MPX 5550/5650 appliance:

- 160 GB removable solid-state drive, which is used to store the NetScaler software and the user data.
- Power switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- USB port (reserved for a future release).
- Non-maskable interrupt (NMI) button, which is used at the request of Technical Support to produce a NetScaler core dump. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Single power supply, rated at 300 watts, 110-220 volts.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

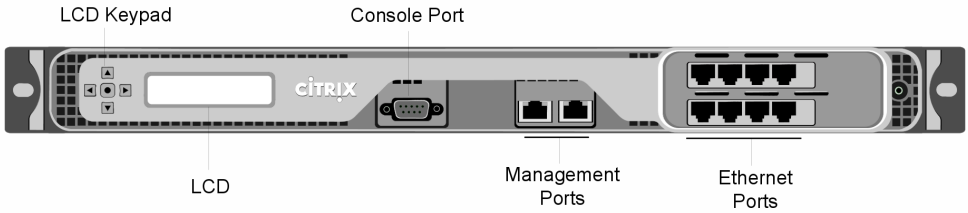
Citrix NetScaler MPX 7500 and MPX 9500

Sep 04, 2013

The Citrix NetScaler MPX 7500/9500 are 1U appliances, each with 1 quad-core processor, and 8 gigabytes (GB) of memory. The MPX 7500/9500 appliances are available in two port configurations: 8x10/100/1000Base-T copper Ethernet ports and 4x1G SFP + 4x10/100/1000Base-T copper Ethernet ports.

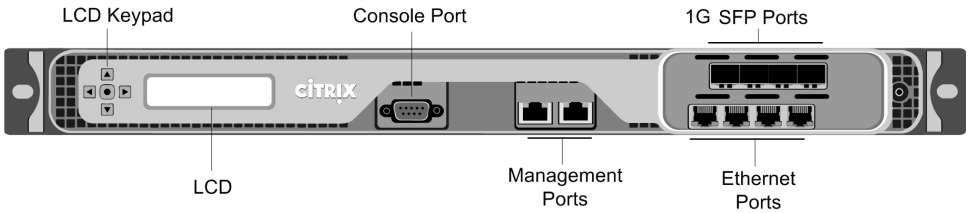
The following figure shows the front panel of the MPX 7500/9500 (8x10/100/1000Base-T copper Ethernet ports) appliances.

Figure 1. Citrix NetScaler MPX 7500/9500 (8x10/100/1000Base-T copper Ethernet ports), front panel



The following figure shows the front panel of the MPX 7500/9500 (4x1G SFP + 4x10/100/1000Base-T copper Ethernet ports) appliances.

Figure 2. Citrix NetScaler MPX 7500/9500 (4x1G SFP + 4x10/100/1000Base-T copper Ethernet ports), front panel

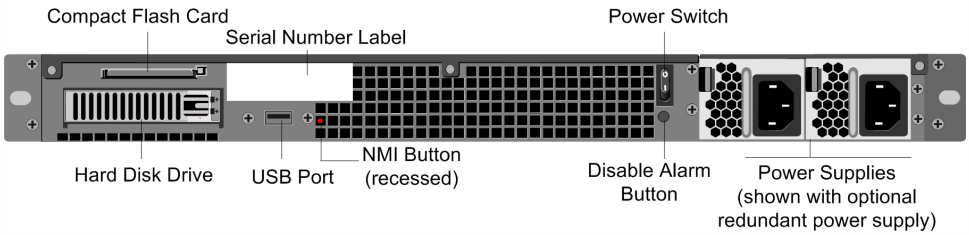


Depending on the model, the appliance has the following ports:

- RS232 serial console port.
- Two 10/100/1000Base-T copper Ethernet management ports, numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Network Ports
  - MPX 7500/9500 (8x10/100/1000Base-T copper Ethernet ports). Eight 10/100/1000Base-T copper Ethernet ports numbered 1/1, 1/2, 1/3, and 1/4 on the top row from left to right, and 1/5, 1/6, 1/7, and 1/8 on the bottom row from left to right.
  - MPX 7500/9500 (4x1G SFP + 4x10/100/1000Base-T copper Ethernet ports). Four 1-gigabit copper or fiber 1G SFP ports numbered 1/1, 1/2, 1/3, and 1/4 on the top row from left to right, and four 10/100/1000BASE-T copper Ethernet ports (RJ45) numbered 1/5, 1/6, 1/7, and 1/8 on the bottom row from left to right.

The following figure shows the back panel of the MPX 7500/9500 appliance.

Figure 3. Citrix NetScaler MPX 7500/9500, back panel



The following components are visible on the back panel of the MPX 7500/9500:

- Four-gigabyte removable CompactFlash card that is used to store the NetScaler software.
- Power switch, which turns off power to the MPX 7500/9500, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Removable hard-disk drive (HDD) that is used to store user data. Appliances shipped before February, 2012 store user data on a HDD. In appliances shipped after February, 2012, a solid-state drive replaces the HDD. Both types of drive have the same functionality and support the same software releases.
- USB port (reserved for a future release).
- Non-maskable interrupt (NMI) button that is used at the request of Technical Support and produces a core dump on the appliance. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Disable alarm button. This button is functional only when the appliance has two power supplies.
  Press this button to stop the power alarm from sounding when you have plugged the MPX 7500/9500 into only one power outlet or when one power supply is malfunctioning and you want to continue operating the MPX 7500/9500 until it is repaired.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

Citrix NetScaler MPX 8005,MPX 8015,MPX 8200, MPX 8400, MPX 8600, and MPX 8800

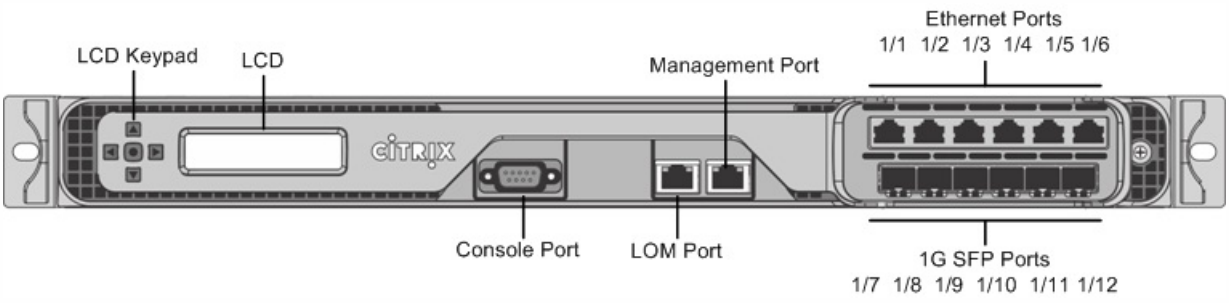
Jan 21, 2014

The Citrix NetScaler models MPX 8005, MPX 8015, MPX 8200, MPX 8400, MPX 8600, and MPX 8800 are 1U appliances. Each model has one quad-core processor and 32 gigabytes (GB) of memory. The MPX 8005/8015/8200/8400/8600/8800 appliances are available in two port configurations:

- Six 10/100/1000Base-T copper Ethernet ports and six 1G SFP ports (6x10/100/1000Base-T copper Ethernet ports + 6x1G SFP)
- Six 10/100/1000Base-T copper Ethernet ports and two 10G SFP+ ports (6x10/100/1000Base-T copper Ethernet ports + 2x10G SFP+)

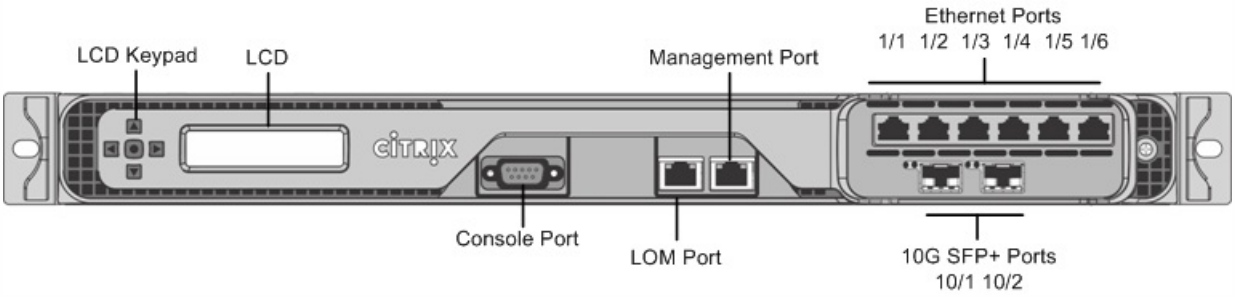
The following figure shows the front panel of the MPX 8005/8015/8200/8400/8600/8800 (6x10/100/1000Base-T copper Ethernet ports + 6x1G SFP) appliance.

Figure 1. Citrix NetScaler MPX 8005/8015/8200/8400/8600/8800 (6x10/100/1000Base-T copper Ethernet ports + 6x1G SFP), front panel



The following figure shows the front panel of the MPX 8005/8015/8200/8400/8600/8800 (6x10/100/1000Base-T copper Ethernet ports + 2x10G SFP+) appliance.

Figure 2. Citrix NetScaler MPX 8005/8015/8200/8400/8600/8800 (6x10/100/1000Base-T copper Ethernet ports + 2x10G SFP+), front panel



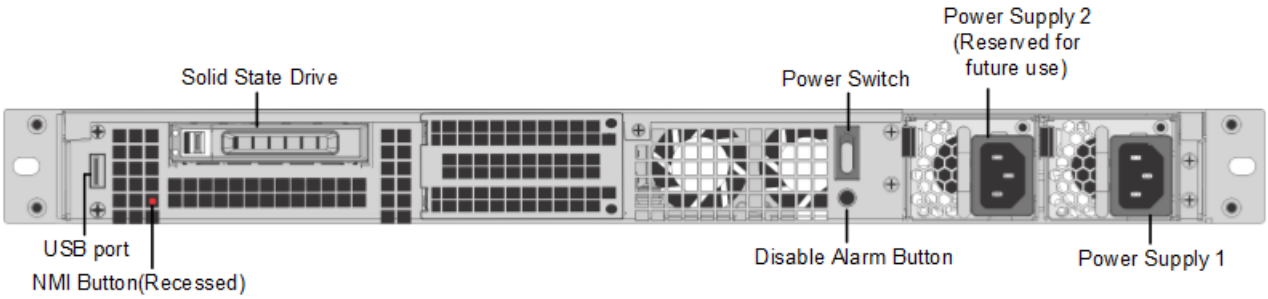
Depending on the model, the appliance has the following ports:

- RS232 serial console port.
- One 10/100Base-T copper Ethernet Port (RJ45), also called LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.

- One 10/100/1000Base-T copper Ethernet management port (RJ45), numbered 0/1. The management port is used to connect directly to the appliance for system administration functions.
- Network Ports
  - MPX 8005/8015/8200/8400/8600/8800 (6x10/100/1000Base-T copper Ethernet ports + 6x1G SFP). Six 10/100/1000BASE-T copper Ethernet ports (RJ45) numbered 1/1, 1/2, 1/3, 1/4, 1/5, and 1/6 on the top row from left to right, and six 1-gigabit copper or fiber 1G SFP ports numbered 1/7, 1/8, 1/9, 1/10, 1/11, and 1/12 on the bottom row from left to right.
  - MPX 8005/8015/8200/8400/8600/8800 (6x10/100/1000Base-T copper Ethernet ports + 2x10G SFP+). Six 10/100/1000BASE-T copper Ethernet ports (RJ45) numbered 1/1, 1/2, 1/3, 1/4, 1/5, and 1/6 on the top row from left to right and two 10-gigabit SFP+ ports numbered 10/1 and 10/2 on the bottom row from left to right.

The following figure shows the back panel of the MPX 8005/8015/8200/8400/8600/8800 appliance.

Figure 3. Citrix NetScaler MPX 8005/8015/8200/8400/8600/8800 appliance, back panel



The following components are visible on the back panel of the MPX 8005/8015/8200/8400/8600/8800 appliance:

- One 256 GB removable solid-state drive, which is used to store the NetScaler software and the user data.

  Note: Earlier MPX 8005/8015/8200/8400/8600/8800 appliances had three additional SSD slots for future use. Current NetScaler MPX 8005/8015/8200/8400/8600/8800 appliances do not have any additional SSD slots for future use.

- Power switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- USB port (reserved for a future release).
- Non-maskable interrupt (NMI) button, which is used at the request of Technical Support to produce a NetScaler core dump. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Disable alarm button, which is nonfunctional. This button is functional only if you install a second power supply. Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet, or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Single power supply, rated at 450 watts, 110-220 volts.

  Note: The MPX 8005/8015/8200/8400/8600/8800 appliance supports dual power supplies, but ships with one power supply. Contact your Citrix sales representative to order a second power supply.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

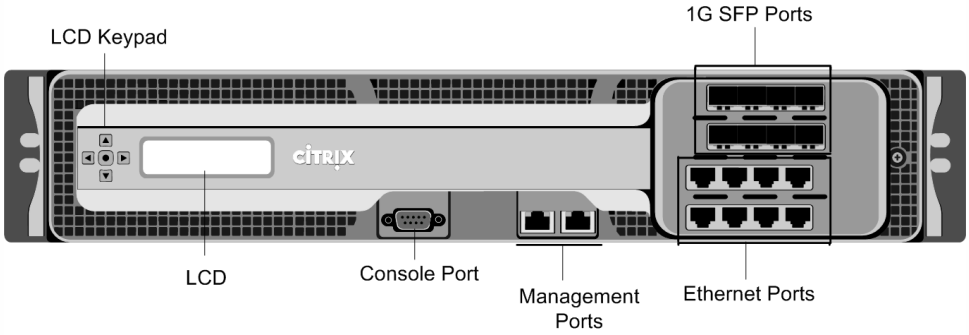
Citrix NetScaler MPX 9700, MPX 10500, MPX 12500, and MPX 15500

Sep 04, 2013

The Citrix NetScaler MPX 9700/10500/12500/15500 are 2U appliances, each with 2 quad-core processors, and 16 gigabytes (GB) of memory. All these appliances are also available in a 10G model and a FIPS model.

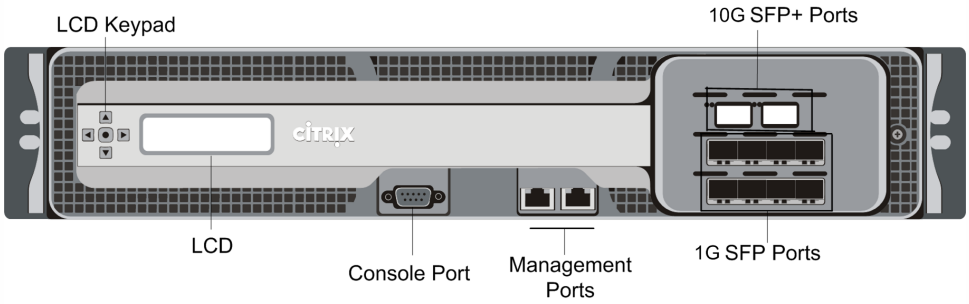
The following figure shows the front panel of the MPX 9700/10500/12500/15500.

Figure 1. Citrix NetScaler MPX 9700/10500/12500/15500, front panel



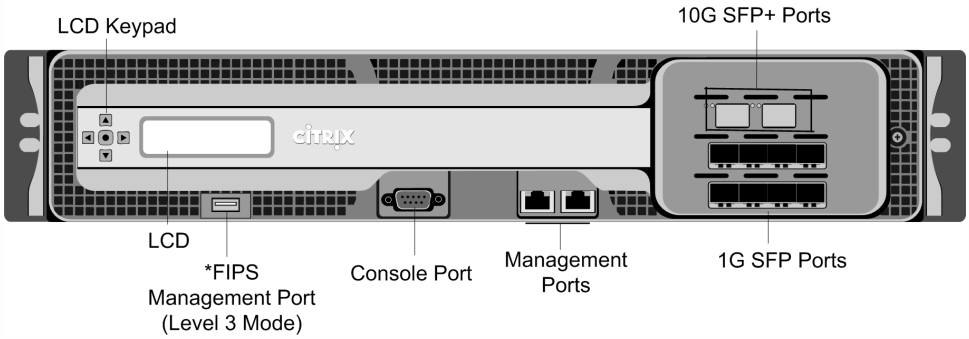
The following figure shows the front panel of the MPX 9700/10500/12500/15500 10G.

Figure 2. Citrix NetScaler MPX 9700/10500/12500/15500 10G, front panel



The following figure shows the front panel of the MPX 9700/10500/12500/15500 FIPS.

Figure 3. Citrix NetScaler MPX 9700/10500/12500/15500 FIPS, front panel



*The FIPS Management Port (Level 3 Mode) is reserved for a future release.

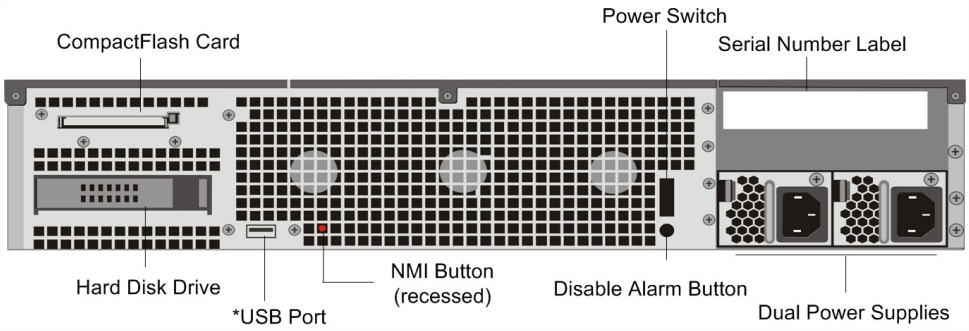
Caution: Do not insert a USB device into the FIPS Management Port. This will cause the FIPS card to fail.

Depending on the model, the appliance has the following ports:

- FIPS Management Port (reserved for a future release).
- RS232 serial Console Port.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Network Ports
  - MPX 9700/10500/12500/15500. Eight copper or fiber 1G SFP ports numbered 1/1, 1/2, 1/3, and 1/4 on the first row from left to right, and 1/5, 1/6, 1/7, and 1/8 on the second row from left to right. Eight 10/100/1000BASE-T copper Ethernet Ports (RJ45) numbered 1/9, 1/10, 1/11, and 1/12 on the third row from left to right, and 1/13, 1/14, 1/15, and 1/16 on the fourth row from left to right.
  - MPX 9700/10500/12500/15500 10G and MPX 9700/10500/12500/15000 FIPS. Two 10G SFP+ Ports numbered 10/1 and 10/2 on the top row, eight 1-gigabit copper or fiber 1G SFP Ports numbered 1/1, 1/2, 1/3, and 1/4 on the middle row from left to right, and 1/5, 1/6, 1/7, and 1/8 on the bottom row from left to right.

Important: The 10-gigabit ports on this appliance are labeled 10/1 and 10/2.
The following figure shows the back panel of the MPX 9700/10500/12500/15500 appliances, including the 10G model and FIPS model.

Figure 4. Citrix NetScaler MPX 9700/10500/12500/15500, MPX 9700/10500/12500/15500 FIPS, and MPX 9700/10500/12500/15500 10G, back panel



*The USB Port is reserved for a future release.

The following components are visible on the back panel of the MPX 9700/10500/12500/15500, including the 10G model and FIPS model:

- Four GB removable CompactFlash Card that is used to store the NetScaler software.
- Power Switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Removable Hard Disk Drive that is used to store user data.
- USB Port (reserved for a future release).
- Non-maskable interrupt (NMI) Button that is used at the request of Technical Support and produces a core dump on the NetScaler. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Disable Alarm Button. This button is functional only when the appliance has two power supplies.
  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Dual Power Supplies, each rated at 450 watts, 110-220 volts.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the

Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

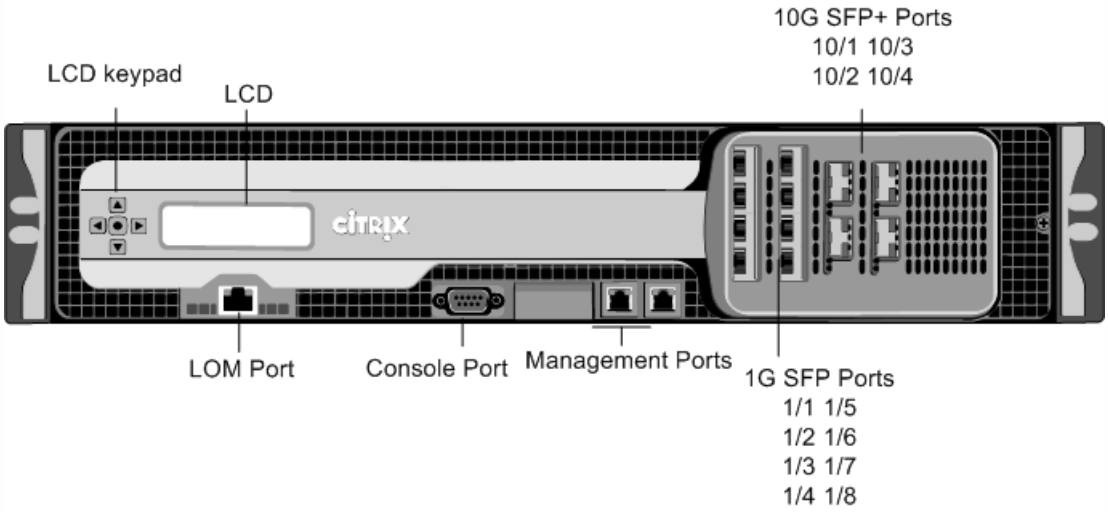
Citrix NetScaler MPX 11500,MPX 13500,MPX 14500,MPX 16500,MPX 18500, andMPX 20500

Oct 25, 2013

The Citrix NetScaler models MPX 11500/13500/14500/16500/18500/20500 are 2U appliances. Each model has two 6-core processors for a total of 12 physical cores (24 cores with hyper-threading), and 48 gigabytes (GB) of memory.

The following figure shows the front panel of the MPX 11500/13500/14500/16500/18500/20500 appliance.

Figure 1. Citrix NetScaler MPX 11500/13500/14500/16500/18500/20500 appliance, front panel

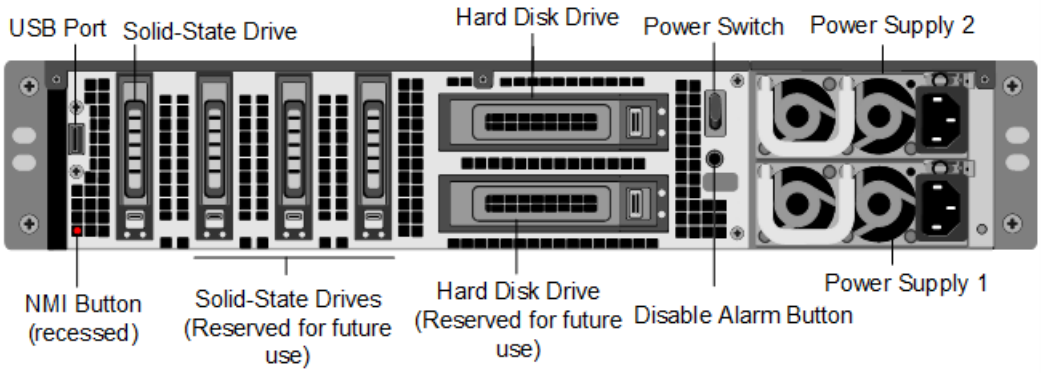


The MPX 11500/13500/14500/16500/18500/20500 appliances have the following ports:

- 10/100Base-T copper Ethernet Port (RJ45), also called LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
  Note: The LEDs on the LOM port are not operational by design.
- RS232 serial console port.
- Two 10/100/1000Base-T copper Ethernet management ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Eight 1G SFP ports numbered 1/1, 1/2, 1/3, 1/4 from top to bottom in the first column, and 1/5, 1/6, 1/7, and 1/8 from top to bottom in the second column.
- Four 10G SFP+ ports numbered 10/1 and 10/2 from top to bottom in the first column, and 10/3 and 10/4 from top to bottom in the second column.

The following figure shows the back panel of the MPX 11500/13500/14500/16500/18500/20500 appliance.

Figure 2. Citrix NetScaler MPX 11500/13500/14500/16500/18500/20500 appliance, back panel



The following components are visible on the back panel of the MPX 11500/13500/14500/16500/18500/20500 appliance:

- 160 GB removable solid-state drive that is used to store the NetScaler software.
- USB port (reserved for a future release).
- Power switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Non-maskable interrupt (NMI) Button that is used at the request of Technical Support and produces a core dump on the NetScaler. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Two removable hard-disk drives that are used to store user data.
- Disable alarm button. This button is functional only when the appliance has two power supplies.
  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Dual power supplies, each rated at 650 watts, 110-220 volts.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

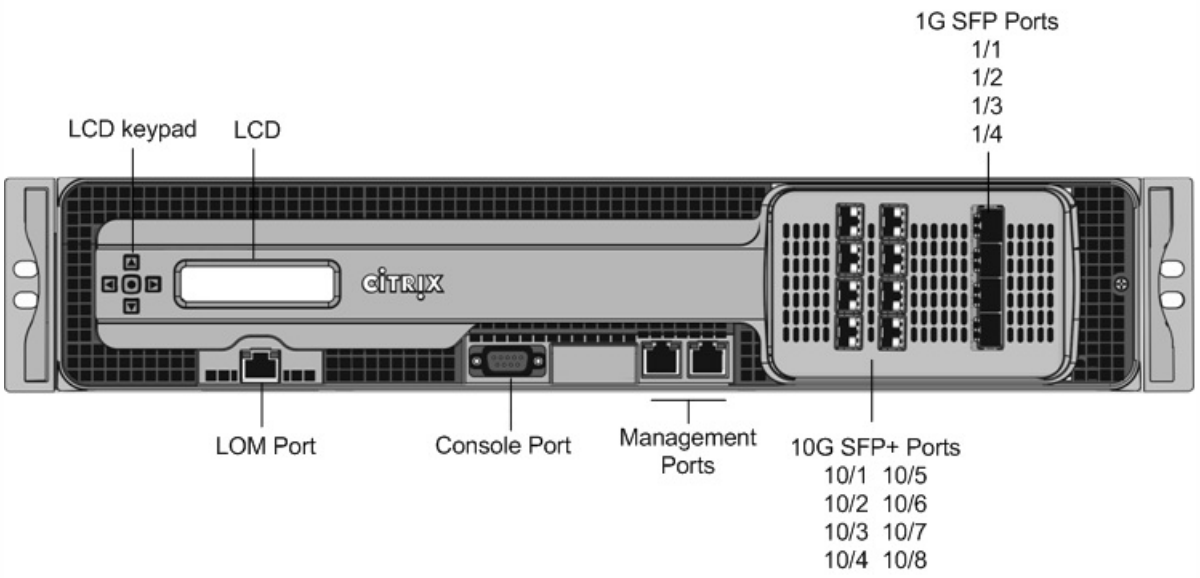
For information about performing initial configuration of your appliance, see "Initial Configuration."

Citrix NetScaler MPX 11515, MPX 11520, MPX 11530, MPX 11540, and MPX 11542

Mar 14, 2014

The Citrix NetScaler models MPX 11515/11520/11530/11540/11542 are 2U appliances. Each model has two 6-core processors for a total of 12 physical cores (24 cores with hyper-threading), and 48 gigabytes (GB) of memory.
The following figure shows the front panel of the MPX 11515/11520/11530/11540/11542 appliance.

Figure 1. Citrix NetScaler MPX 11515/11520/11530/11540/11542 appliance, front panel

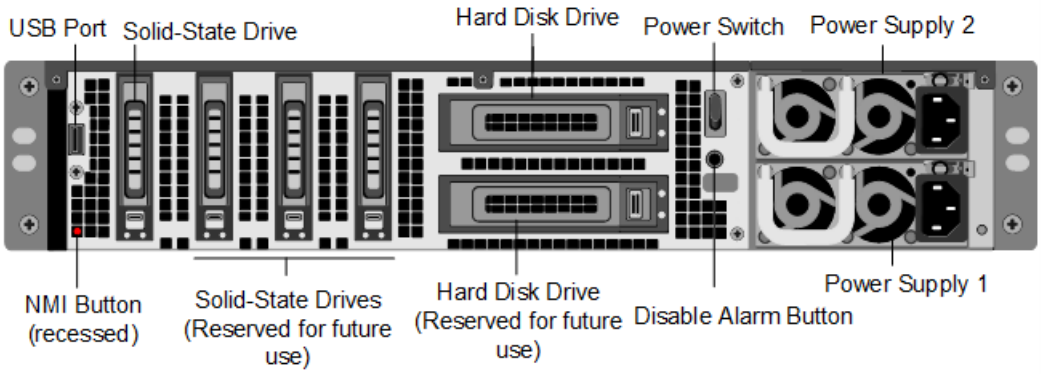


The MPX 11515/11520/11530/11540/11542 appliances have the following ports:

- RS232 serial console port.
- 10/100Base-T copper Ethernet Port (RJ45), also called LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
  Note: The LEDs on the LOM port are not operational by design.
- Two 10/100/1000Base-T copper Ethernet management ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Eight 10G SFP+ ports and four copper or fiber 1G SFP ports.

The following figure shows the back panel of the MPX 11515/11520/11530/11540/11542 appliance.

Figure 2. Citrix NetScaler MPX11515/11520/11530/11540/11542 appliance, back panel



The following components are visible on the back panel of the MPX 11515/11520/11530/11540/11542 appliance:

- 256 GB removable solid-state drive that is used to store the NetScaler software.
- USB port (reserved for a future release).
- Power switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Non-maskable interrupt (NMI) Button that is used at the request of Technical Support and produces a core dump on the NetScaler. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Two removable hard-disk drives that are used to store user data.
- Disable alarm button. This button is functional only when the appliance has two power supplies.
  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Dual power supplies, each rated at 650 watts, 110-220 volts.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

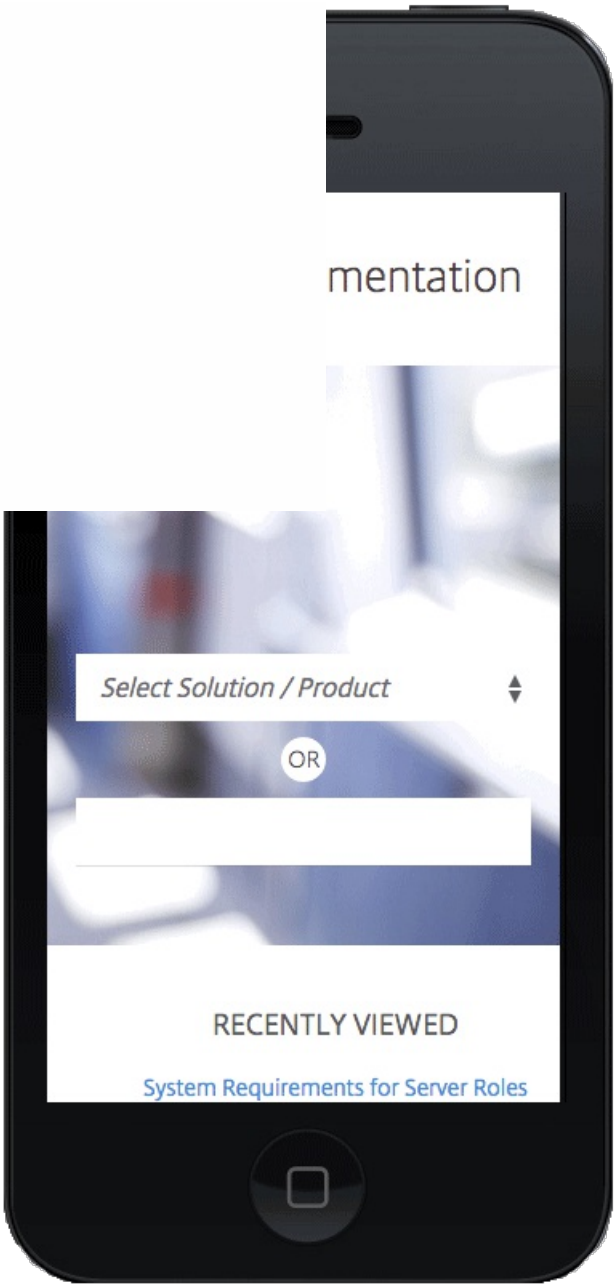
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it

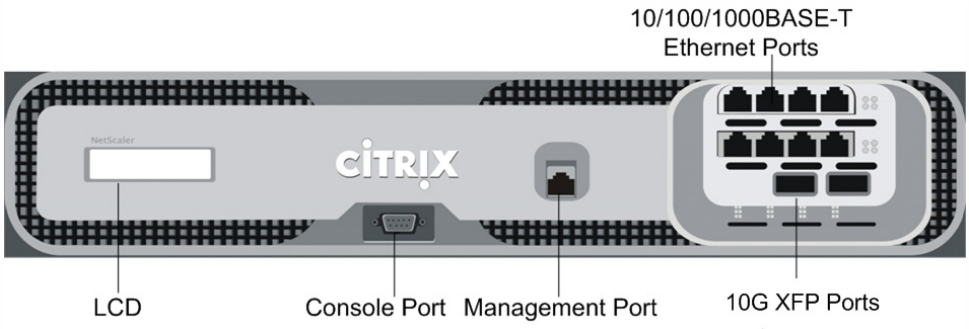


Citrix NetScaler MPX 15000

Sep 04, 2013

The Citrix NetScaler MPX 15000 appliance is a 2U appliance, with 2 dual-core processors, and 16 GB of memory. The MPX 15000 is a high-capacity hardware platform intended for heavy use in enterprise and service provider environments. The following figure shows the front panel of the MPX 15000 appliance.

Figure 1. Citrix NetScaler MPX 15000 appliance, front panel



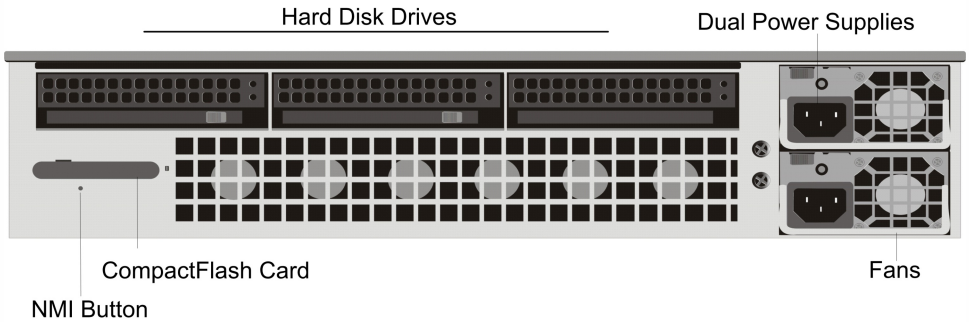
The appliance has the following ports:

- RS232 serial console port.
- 10/100/1000BASE-T copper Ethernet management port, numbered 0/1.
- Two XFP (10-Gigabit Small Form-Factor Pluggable) fiber optic ports, numbered from left to right 1/1 and 1/2.
- Eight 10/100/1000BASE-T copper Ethernet ports, numbered from upper left to bottom right 1/3, 1/4, 1/5, 1/6, 1/7, 1/8, 1/9, and 1/10.

When facing the bezel, the upper LEDs to the left of each port represent connectivity. They are lit and amber in color when active. The lower LEDs represent throughput. They are lit and green when active.

Note: The network port numbers on all appliances consist of two numbers separated by a forward slash. The first number is the port adapter slot number and will always be either 0 or 1. The second number is the interface port number. Ports on appliances are numbered sequentially starting with 1.
The following figure shows the back panel of the MPX 15000 appliance.

Figure 2. Citrix NetScaler MPX 15000 appliance, back panel



The following components are visible on the back panel of the MPX 15000 appliance:

- Removable hard-disk drive that is used to store user data.
- Dual power supplies, each rated at 500 watts, 110-220 volts.
  You plug separate power cords into the power supplies and connect them to separate wall sockets. The MPX 15000 functions properly with a single power supply; the extra power supply serves as a backup.

- Non-maskable interrupt (NMI) button, which signals the MPX 15000 to perform an orderly shutdown after saving all files. You must use a pen, pencil, or other pointed object to press this button, which is located inside a small hole to prevent it from being pressed accidentally.
- Removable CompactFlash card that is used to store the NetScaler software.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

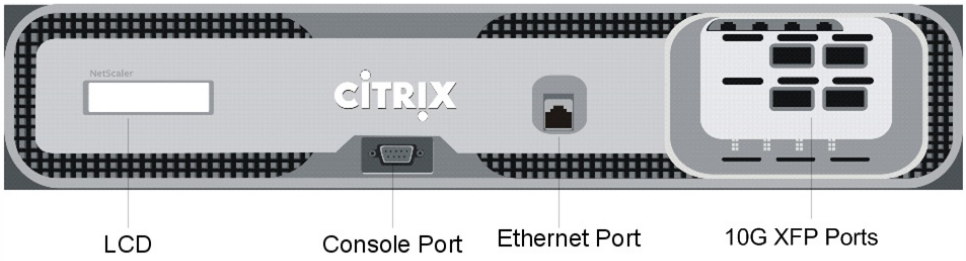
Citrix NetScaler MPX 17000

Sep 04, 2013

The Citrix NetScaler MPX 17000 appliance is a 2U appliance, with 2 quad-core processors, and 32 GB of memory. The MPX 17000 is a high-capacity hardware platform intended for any high traffic, intensive processing data center environment. There are two MPX 17000 models: the four network-port model and the ten network-port model.

The following figure shows the front panel of the MPX 17000, four network-port model.

Figure 1. Citrix NetScaler MPX 17000 four network-port model, front panel



Depending on the model, the appliance has the following ports:

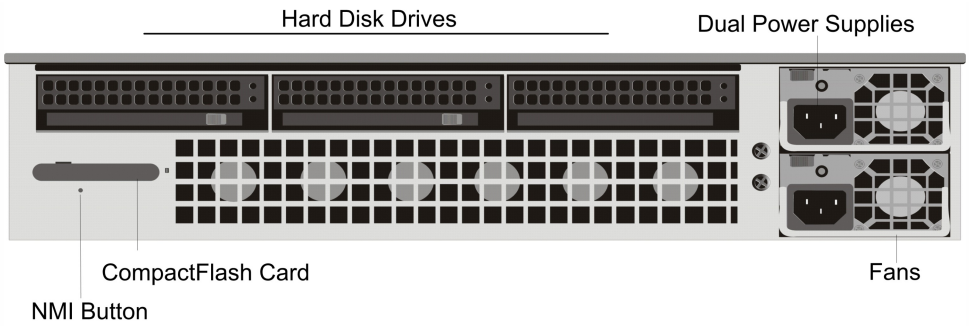
- RS232 serial console port.
- 10/100/1000BASE-T copper Ethernet management port, numbered 0/1.
- Network Ports
  - MPX 17000 four network-port model. Four XFP (10-Gigabit Small Form-Factor Pluggable) ports, numbered from upper left to bottom right 1/1, 1/2, 1/3, and 1/4.
  - MPX 17000 ten network-port model. Two XFP ports, numbered from left to right 1/1 and 1/2 and eight 10/100/1000BASE-T Ethernet ports, numbered from upper left to bottom right 1/3, 1/4, 1/5, 1/6, 1/7, 1/8, 1/9 and 1/10.

  Note: The network port numbers on all appliances consist of two numbers separated by a forward slash. The first number is the port adapter slot number and will always be either 0 or 1. The second number is the interface port number. Ports on appliances are numbered sequentially starting with 1.

When facing the bezel, the upper LEDs to the left of each port represent connectivity. They are lit and amber in color when active. The lower LEDs represent throughput. They are lit and green when active.

The following figure shows the back panel of the MPX 17000 appliance.

Figure 2. Citrix NetScaler MPX 17000 appliance, back panel



The following components are visible on the back of the MPX 17000 appliance:

- Removable hard-disk drive that is used to store user data.
- Dual power supplies, each rated at 500 watts, 110-220 volts.
  You plug separate power cords into the power supplies and connect them to separate wall sockets. The MPX 17000 functions properly with a single power supply; the extra power supply serves as a backup.
- Non-maskable interrupt (NMI) button, which signals the MPX 17000 to perform an orderly shutdown after saving all files. You must use a pen, pencil, or other pointed object to press this button, which is located inside a small hole to prevent it from being pressed accidentally.
- Removable CompactFlash card that is used to store the NetScaler software.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

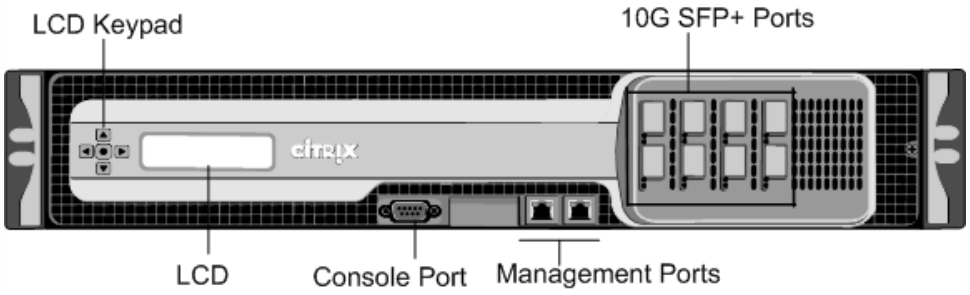
Citrix NetScaler MPX 17500, MPX 19500, and MPX 21500

Oct 25, 2013

The Citrix NetScaler models MPX 17500/19500/21500 are 2U appliances. Each model has two 6-core processors and 48 gigabytes (GB) of memory.

The following figure shows the front panel of the MPX 17500/19500/21500 appliance.

Figure 1. Citrix NetScaler MPX 17500/19500/21500 appliance, front panel

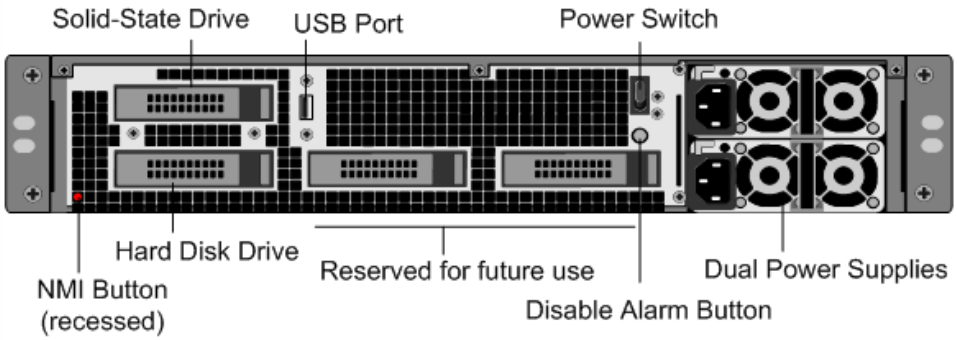


The MPX 17500/19500/21500 appliances have the following ports:

- RS232 serial console port.
- Two 10/100/1000Base-T copper Ethernet management ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Eight 10G SFP+ ports numbered 10/1, 10/2, 10/3, and 10/4 on the top row from left to right, and 10/5, 10/6, 10/7, and 10/8 on the bottom row from left to right.

The following figure shows the back panel of the MPX 17500/19500/21500 appliance.

Figure 2. Citrix NetScaler MPX 17500/19500/21500 appliance, back panel



The following components are visible on the back panel of the MPX 17500/19500/21500 appliance:

- 160 GB removable solid-state drive that is used to store the NetScaler software.
- USB port (reserved for a future release).
- Power switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Non-maskable interrupt (NMI) button that is used at the request of Technical Support and produces a core dump on the NetScaler. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent

unintentional activation.

- Removable hard-disk drive that stores user data.
- Disable alarm button. This button is functional only when the appliance has two power supplies.
  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Dual power supplies, each rated at 650 watts, 110-220 volts.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

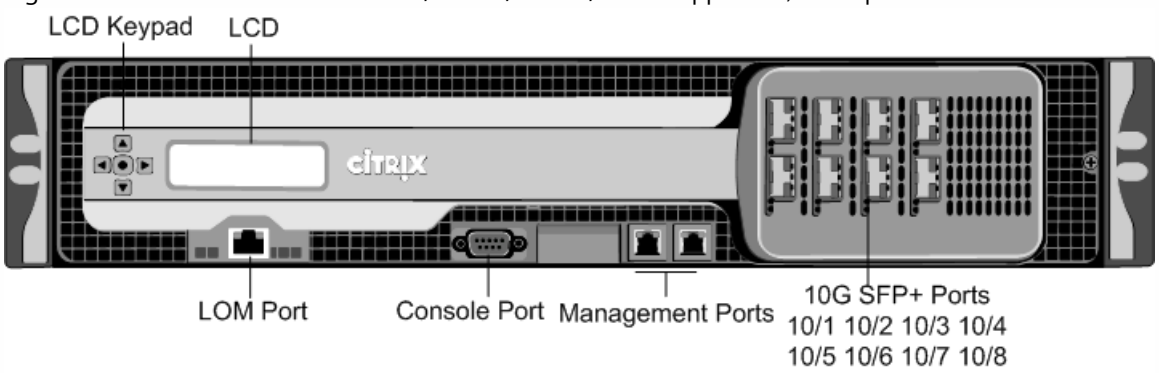
Citrix NetScaler MPX 17550, MPX 19550, MPX 20550, and MPX 21550

Oct 25, 2013

The Citrix NetScaler models MPX 17550, MPX 19550, MPX 20550, and MPX 21550 are 2U appliances. Each model has two 6-core processors for a total of 12 physical cores (24 cores with hyper-threading), and 96 gigabytes (GB) of memory.

The following figure shows the front panel of the MPX 17550/19550/20550/21550 appliance.

Figure 1. Citrix NetScaler MPX 17550/19550/20550/21550 appliance, front panel

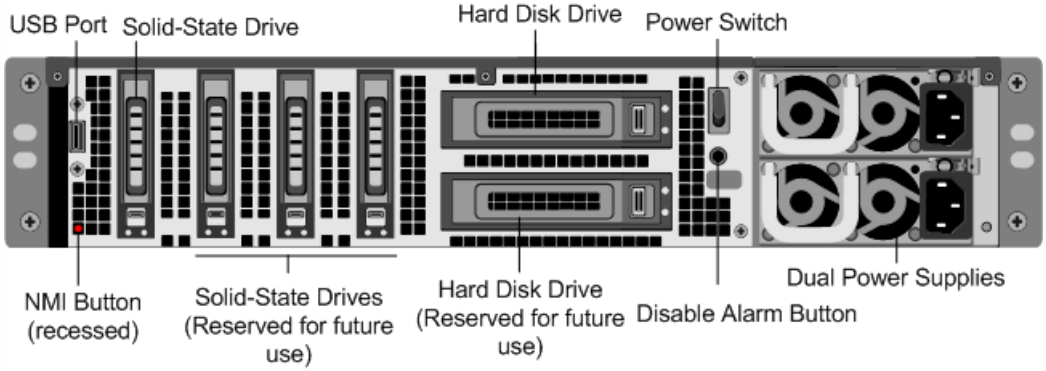


The MPX 17550/19550/20550/21550 appliance has the following ports:

- 10/100Base-T copper Ethernet Port (RJ45), also called LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
  Note: The LEDs on the LOM port are not operational by design.
- RS232 serial console port.
- Two 10/100/1000Base-T copper Ethernet management ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Eight 10G SFP+ ports numbered 10/1, 10/2, 10/3, and 10/4 on the top row from left to right, and 10/5, 10/6, 10/7, and 10/8 on the bottom row from left to right.

The following figure shows the back panel of the MPX 17550/19550/20550/21550 appliance.

Figure 2. Citrix NetScaler MPX 17550/19550/20550/21550 appliance, back panel



The following components are visible on the back panel of the MPX 17550/19550/20550/21550 appliance:

- 160 GB removable solid-state drive that is used to store the NetScaler software.
- USB port (reserved for a future release).

- Power switch, which turns off power to the appliance, just as if you were to unplug the power supply. Press the switch for five seconds to turn off the power.
- Non-maskable interrupt (NMI) button that is used at the request of Technical Support and produces a core dump on the NetScaler. You must use a pen, pencil, or other pointed object to press this red button, which is recessed to prevent unintentional activation.
- Two removable hard-disk drives that store user data.
- Disable alarm button. This button is functional only when the appliance has two power supplies.
  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Dual power supplies, each rated at 850 watts, 110-220 volts.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

Citrix NetScaler MPX 22040, MPX 22060, MPX 22080, MPX 22100, and MPX 22120

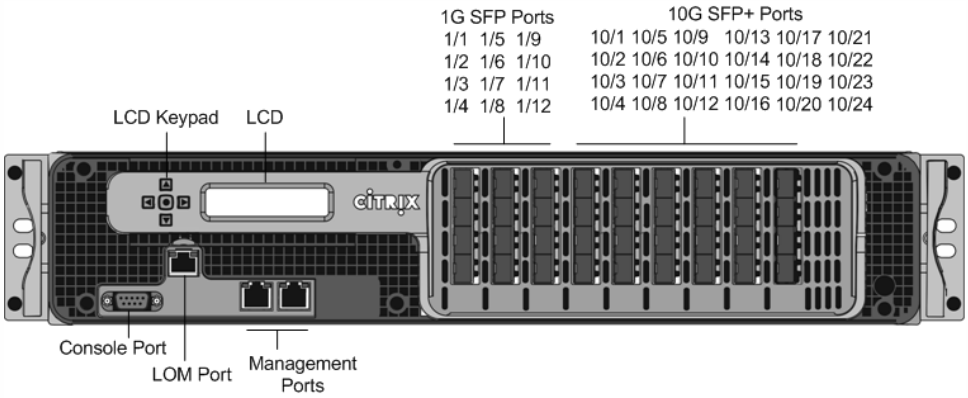
Oct 25, 2013

The Citrix NetScaler MPX 22040/22060/22080/22100/22120 are 2U appliances. Each model has two 8-core processors and 256 gigabytes (GB) of memory. The MPX 22040/22060/22080/22100/22120 appliances are available in two port configurations:

- Twelve 1G SFP ports and twenty-four 10G SFP+ ports (12x1G SFP + 24x10G SFP+)
- Twenty-four 10G SFP+ ports (24x10G SFP+)

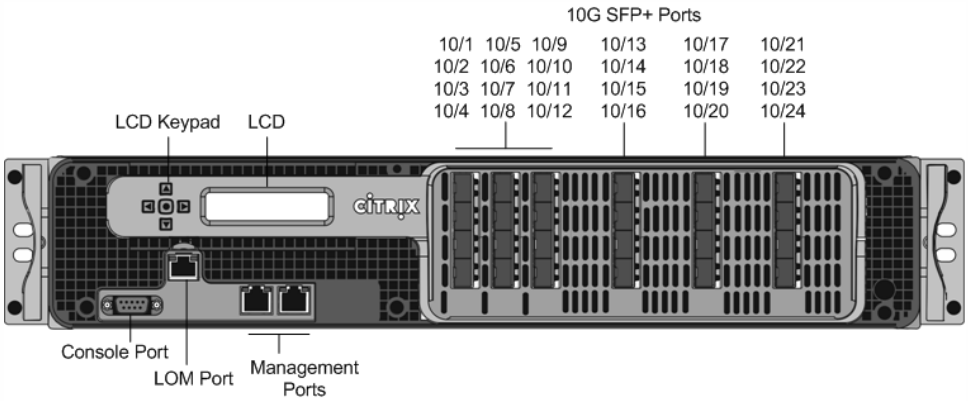
The following figure shows the front panel of the MPX 22040/22060/22080/22100/22120 (12x1G SFP + 24x10G SFP+) appliance.

Figure 1. Citrix NetScaler MPX 22040/22060/22080/22100/22120 (12x1G SFP + 24x10G SFP+), front panel



The following figure shows the front panel of the MPX 22040/22060/22080/22100/22120 (24x10G SFP+) appliance.

Figure 2. Citrix NetScaler MPX 22040/22060/22080/22100/22120 (24x10G SFP+), front panel



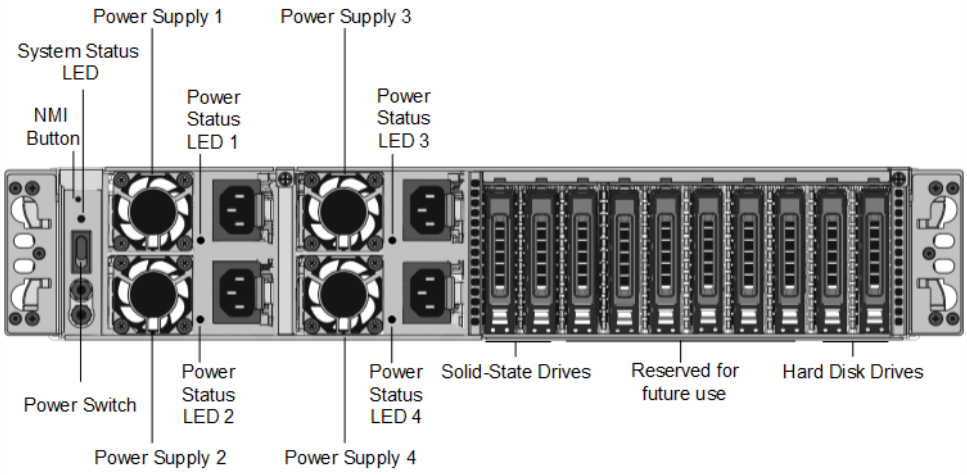
Depending on the model, the appliance has the following ports:

- RS232 serial Console Port.
- 10/100Base-T copper Ethernet Port (RJ45), also called the LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.

- Network Ports
  - MPX 22040/22060/22080/22100/22120 (12x1G SFP + 24x10G SFP+). Twelve copper or fiber 1G SFP ports and twenty-four 10G SFP+ ports.
  - MPX 22040/22060/22080/22100/22120 (24x10G SFP+). Twenty-four 10G SFP+ ports.

The following figure shows the back panel of the MPX 22040/22060/22080/22100/22120 appliances.

Figure 3. Citrix NetScaler MPX 22040/22060/22080/22100/22120, back panel



The following components are visible on the back panel of the MPX 22040/22060/22080/22100/22120 appliance:

- Non-maskable interrupt (NMI) Button, used at the request of Technical Support to initiate a core dump. To press this red button, which is recessed to prevent unintentional activation, use a pen, pencil, or other pointed object. The NMI Button is also available remotely over the network in the LOM GUI, in the Remote Control menu.
- System status LED, which indicates the status of the appliance, as described in http://support.citrix.com/proddocs/topic/netscaler-hrdwre-installation-10-5/ns-hardware-common-components-ref.html.

  Note: On an MPX 22040/22060/22080/22100/22120 appliance running LOM firmware version 3.22, the system status LED indicates an error (continuously glows RED) even though the appliance is functioning properly.
- Four power supplies, each rated at 750 watts, 100-240 volts. A minimum of two power supplies are required for proper operation. The extra power supplies act as backup. Each power supply has an LED that indicates the status of the power supply, as described in http://support.citrix.com/proddocs/topic/netscaler-hrdwre-installation-10-5/ns-hardware-common-components-ref.html.
- Power switch, which turns off power to the appliance. Press the switch for less than two seconds to turn off the power.
- Two 256 GB removable solid-state drives. The leftmost solid-state drive stores the NetScaler software. The other solid-state drive stores user data.
- Two 1TB removable hard disk drives that are used to store user data.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

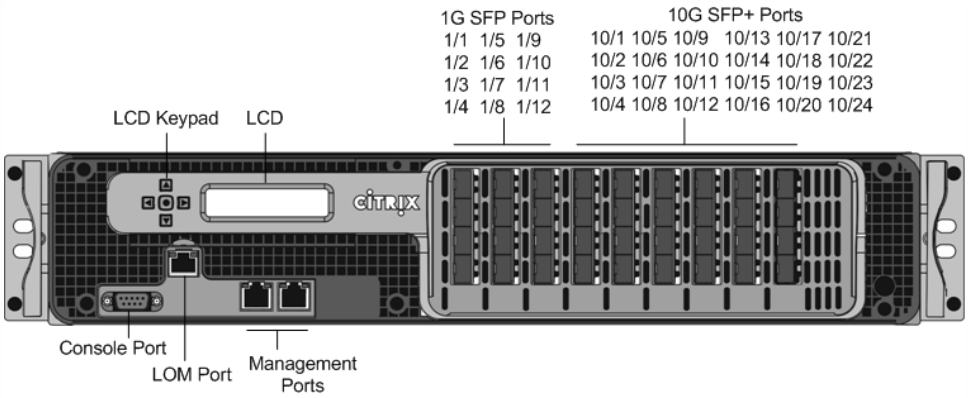
Citrix NetScaler MPX 24100 and MPX 24150

Oct 25, 2013

The Citrix NetScaler MPX 24100/24150 are 2U appliances. Each model has two 8-core processors and 256 gigabytes (GB) of memory. The MPX 24100/24150 appliances are available in the twelve 1G SFP ports and twenty-four 10G SFP+ ports (12x1G SFP + 24x10G SFP+) configuration.

The following figure shows the front panel of the MPX 24100/24150 (12x1G SFP + 24x10G SFP+) appliance.

Figure 1. Citrix NetScaler MPX 24100/24150 (12x1G SFP + 24x10G SFP+), front panel

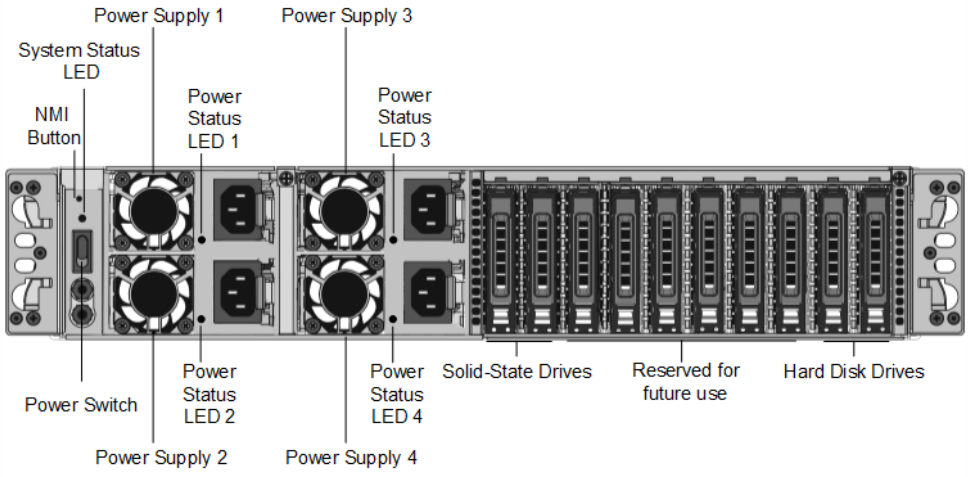


Depending on the model, the appliance has the following ports:

- RS232 serial Console Port.
- 10/100Base-T copper Ethernet Port (RJ45), also called the LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Network Ports
  - MPX 24100/24150 (12x1G SFP + 24x10G SFP+). Twelve copper or fiber 1G SFP ports and twenty-four 10G SFP+ ports.

The following figure shows the back panel of the MPX 24100/24150 appliances.

Figure 2. Citrix NetScaler MPX 24100/24150, back panel



The following components are visible on the back panel of the MPX 24100/24150 appliance:

- Non-maskable interrupt (NMI) Button, used at the request of Technical Support to initiate a core dump. To press this red button, which is recessed to prevent unintentional activation, use a pen, pencil, or other pointed object. The NMI Button is also available remotely over the network in the LOM GUI, in the Remote Control menu.
- System status LED, which indicates the status of the appliance, as described in http://support.citrix.com/proddocs/topic/netscaler-hrdwre-installation-10-5/ns-hardware-common-components-ref.html.

  Note: On an MPX 24100/24150 appliance running LOM firmware version 3.22, the system status LED indicates an error (continuously glows RED) even though the appliance is functioning properly.
- Four power supplies, each rated at 750 watts, 100-240 volts. A minimum of two power supplies are required for proper operation. The extra power supplies act as backup. Each power supply has an LED that indicates the status of the power supply, as described in http://support.citrix.com/proddocs/topic/netscaler-hrdwre-installation-10-5/ns-hardware-common-components-ref.html.
- Power switch, which turns off power to the appliance. Press the switch for less than two seconds to turn off the power.
- Two 128 GB removable solid-state drives.
- One 500 GB removable hard disk drive that is used to store user data.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

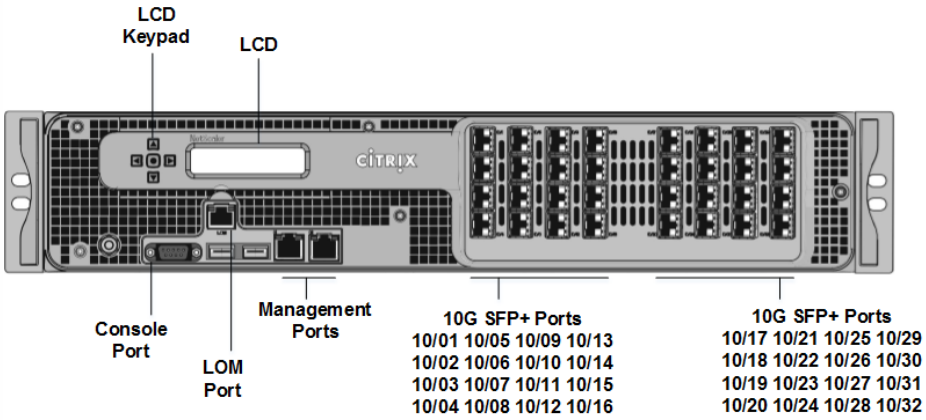
Citrix NetScaler MPX 25100T and MPX 25160T

Oct 25, 2013

The Citrix NetScaler MPX 25100T and 25160T are 2U appliances. Each model has two 10-core processors and 128 gigabytes (GB) of memory. The MPX 25100T/25160T appliances are available in the thirty-two 10G SFP+ ports (32x10G SFP+) configuration.

Note: The MPX 25000T appliances are not RAID (redundant array of independent disks) devices.
The following figure shows the front panel of the MPX 25100T/25160T (32x10G SFP+) appliance.

Figure 1. Citrix NetScaler MPX 25100T/25160T (32x10G SFP+), front panel

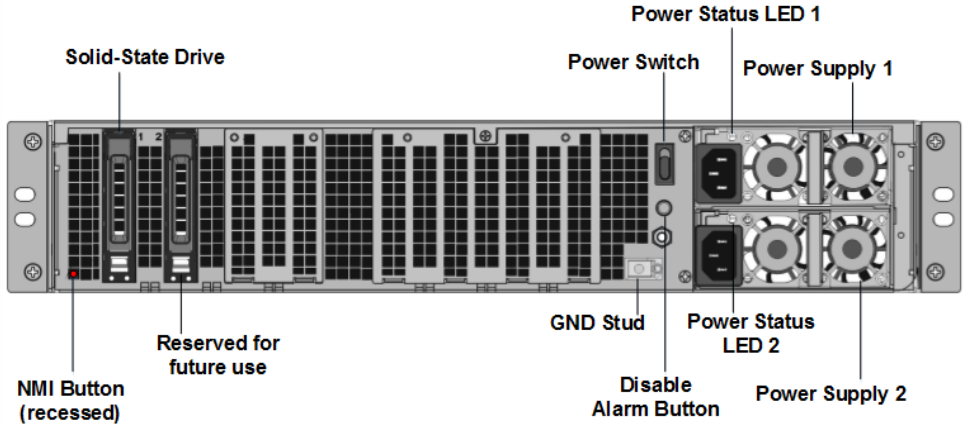


Depending on the model, the appliance has the following ports:

- RS232 serial Console Port.
- 10/100Base-T copper Ethernet Port (RJ45), also called the LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Network Ports, thirty-two 10G SFP+ ports (32x10G SFP+).
  Note: The 10G SFP+ ports on these appliances support copper 1G SFP transceivers.

The following figure shows the back panel of the MPX 25100T/25160T appliance.

Figure 2. Citrix NetScaler MPX 25100T/25160T, back panel



The following components are visible on the back panel of the MPX 25100T/25160T appliance:

- One 300 GB removable solid-state drive.
- Power switch, which turns power to the appliance on or off. Press the switch for less than two seconds to turn off the power.
- Two power supplies, each rated at 1000 watts, 100-240 volts. Each power supply has an LED that indicates the status of the power supply, as described in http://support.citrix.com/proddocs/topic/netscaler-hrdwre-installation-10-1/ns-hardware-common-components-ref.html.
- Disable alarm button, which is functional only when the appliance has two power supplies.

  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet, or when one power supply is malfunctioning, and you want to continue operating the appliance until it is repaired.

- Non-maskable interrupt (NMI) Button, used at the request of Technical Support to initiate a core dump. To press this red button, which is recessed to prevent unintentional activation, use a pen, pencil, or other pointed object. The NMI Button is also available remotely over the network in the LOM GUI, in the Remote Control menu. For more information on the Lights Out Management Port of the appliance, see http://support.citrix.com/proddocs/topic/netscaler-hrdwre-installation-10-1/ns-hardware-lom-intro-wrapper-con.html.

For information about installing the rails, rack mounting the hardware, and connecting the cables, see "Installing the Hardware."

For information about performing initial configuration of your appliance, see "Initial Configuration."

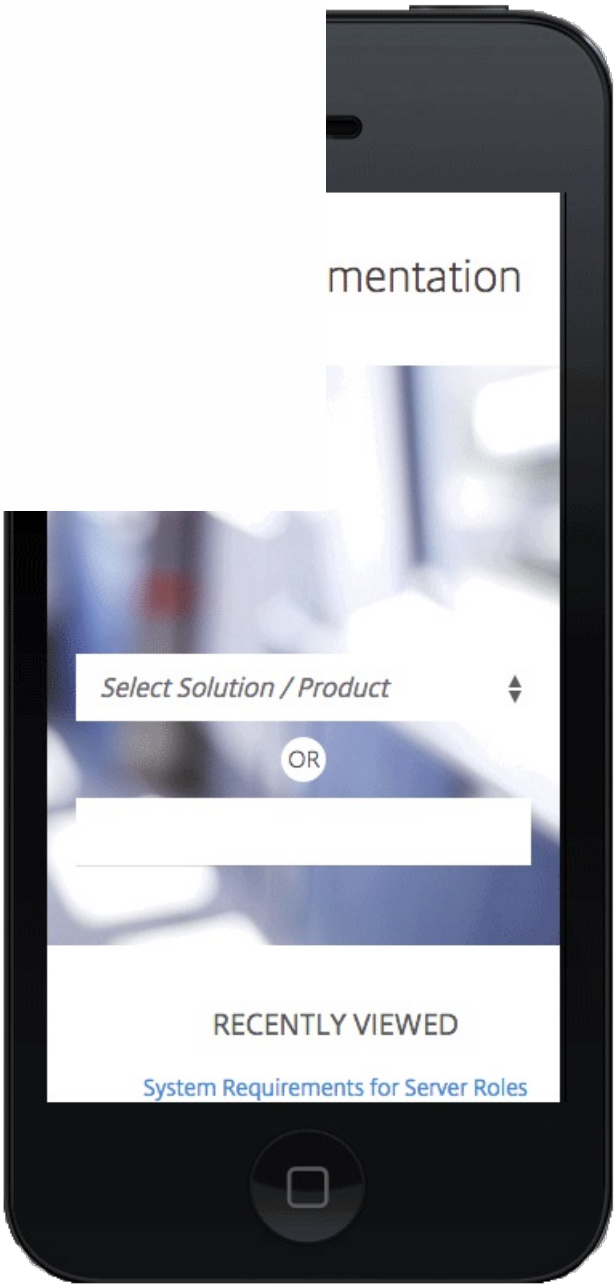
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



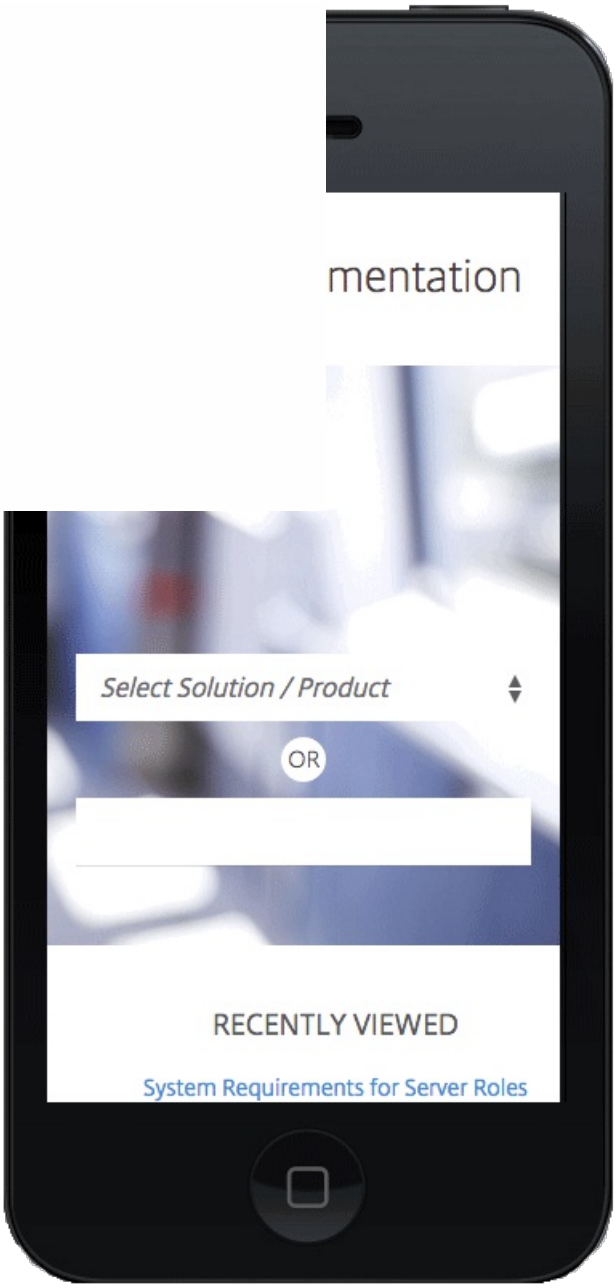
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



Citrix NetScaler MPX 14020, MPX 14030, MPX 14040, MPX 14060, MPX 14080, and MPX 14100

Nov 09, 2015

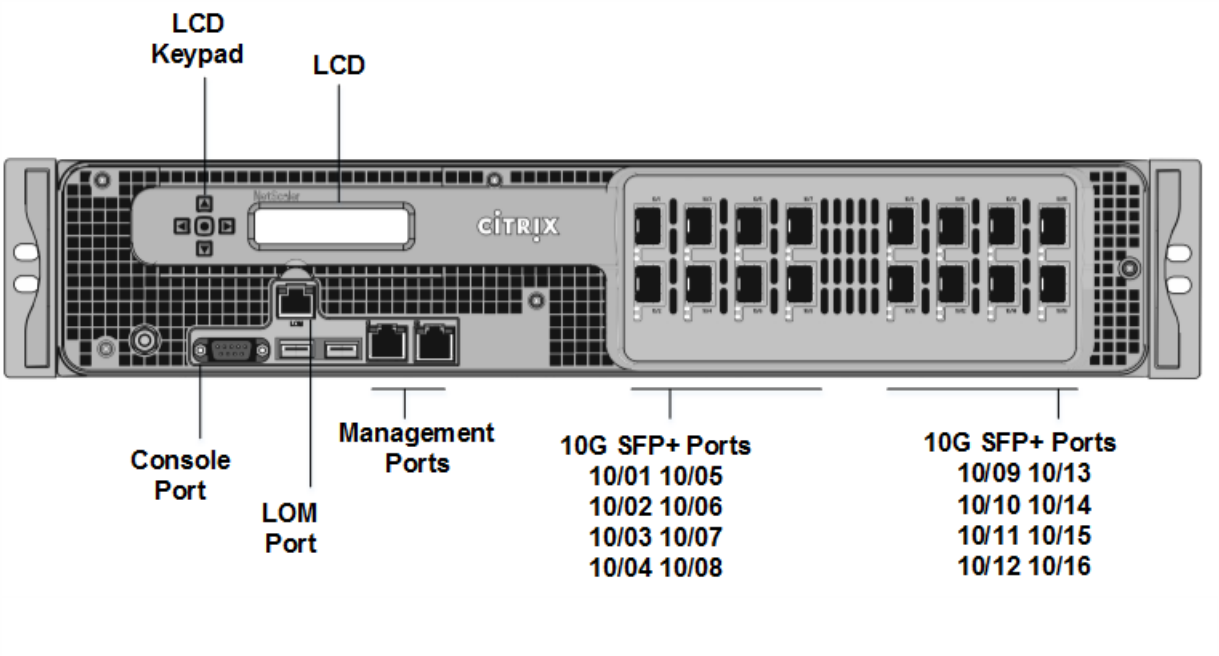
The Citrix NetScaler MPX 14020/14030/14040/14060/14080/14100 are 2U appliances. Each model has two 6-core processors and 64 gigabytes (GB) of memory and sixteen 10G SFP+ ports (16x10G SFP+).

The NetScaler MPX 14020/14030/14040/14060/14080/14100 appliances are shipped with NetScaler release 10.1 Build 133.13 and the LOM version is 4.07.

For information on the software releases supported on the NetScaler hardware platforms, see http://docs.citrix.com/en-us/netscaler/11/getting-started-with-netscaler/product-line.html.

The following figure shows the front panel of the MPX 14020/14030/14040/14060/14080 (16x10G SFP+) appliance.

Figure 1. Citrix NetScaler MPX 14020/14030/14040/14060/14080/14100 (16x10G SFP+), front panel



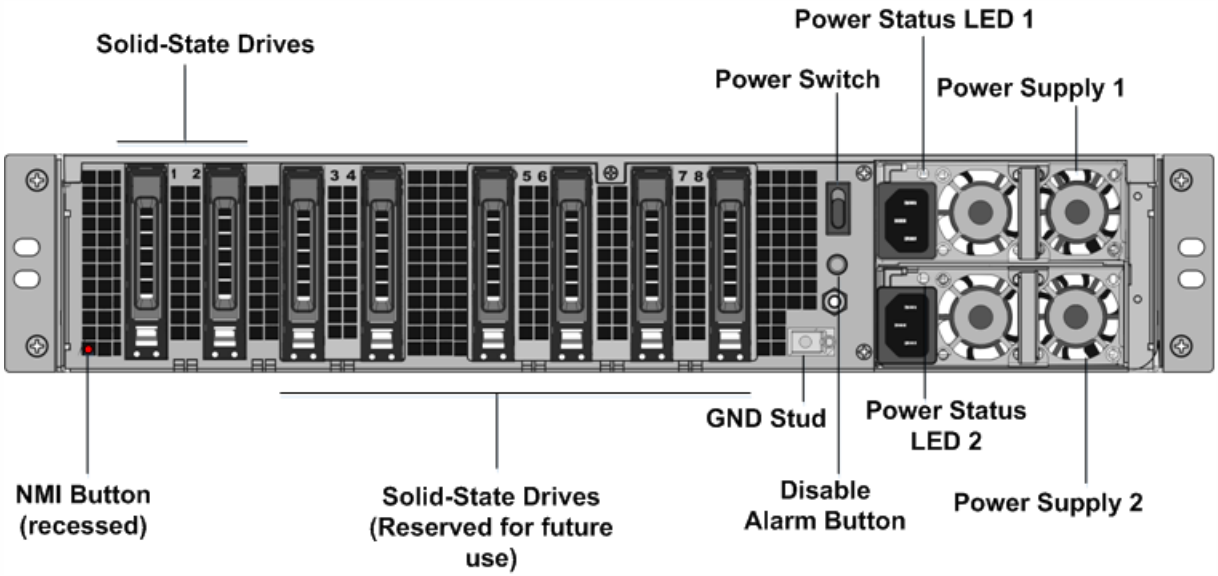
The NetScaler MPX 14020/14030/14040/14060/14080/14100 appliances have the following ports:

- RS232 serial Console Port.
- 10/100Base-T copper Ethernet Port (RJ45), also called the LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1and 0/2 from left to right. These ports are used to connect directly to the appliancefor system administration functions.
- Network Ports, sixteen 10G SFP+ ports (16x10G SFP+).

Note: The 10G SFP+ ports on these appliances support copper 1G SFP transceivers.

The following figure shows the back panel of the MPX 14020/14030/14040/14060/14080/ 14100 appliance.

Figure 2. Citrix NetScaler MPX 14020/14030/14040/14060/14080/14100, back panel



The following components are visible on the back panel of the MPX 14020/14030/14040/14060/14080/14100 appliance:

- Two 240 GB removable solid-state drives (SSDs).

These appliances are redundant array of independent disks (RAID) devices. In a RAID configuration, the same data is stored on multiple drives to improve performance, increase storage capacity, lower the risk of data loss, and provide fault tolerance.

The two SSDs store the same data. If one fails and you replace it, the new SSD mirrors the other one.

- Power switch, which turns power to the appliance on or off. Press the switch for lessthan two seconds to turn off the power.
- Two power supplies, each rated at 1000 watts, 100-240 volts. Each power supply has an LED that indicates the status of the power supply, as described in http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-common-components-ref.html.
- Disable alarm button, which is functional only when the appliance has two powersupplies.

Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet, or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.

- Non-maskable interrupt (NMI) Button, used at the request of Technical Support toinitiate a core dump. To press this red button, which is recessed to preventunintentional activation, use a pen, pencil, or other pointed object. The NMI Buttonis also available remotely over the network in the LOM GUI, in the Remote Controlmenu. For more information about the lights out management port of the appliance, see http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-lom-intro-wrapper-con.html.

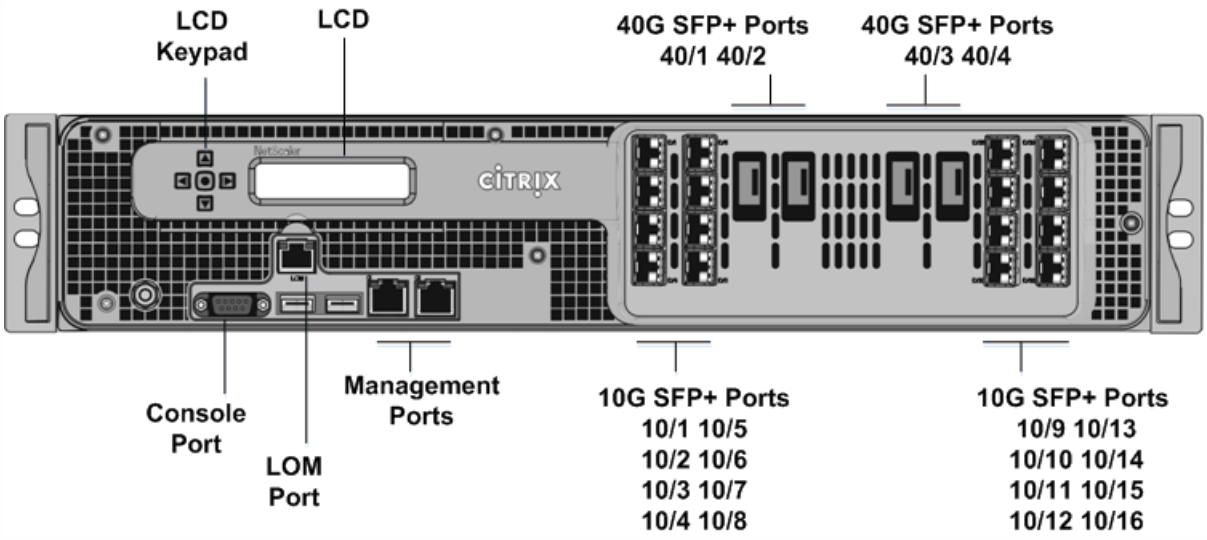
Citrix NetScaler MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G

Mar 24, 2016

The Citrix NetScaler MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G are 2U appliances. Each model has two 6-core processors, 64 gigabytes (GB) of memory, four 40G QSFP+ ports, and sixteen 10G SFP+ ports (4x40G QSFP+ + 16x10G SFP+).

The following figure shows the front panel of the MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G (4x40G QSFP+ + 16x10G SFP+) appliance.

Figure 1. Citrix NetScaler MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G (4x40G QSFP+ + 16x10G SFP+), front panel



The NetScaler MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G appliances have the following ports:

- RS232 serial Console Port.
- 10/100Base-T copper Ethernet Port (RJ45), also called the LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Network Ports, four 40G QSFP+ ports and sixteen 10G SFP+ ports (4x40G QSFP+ + 16x10G SFP+).

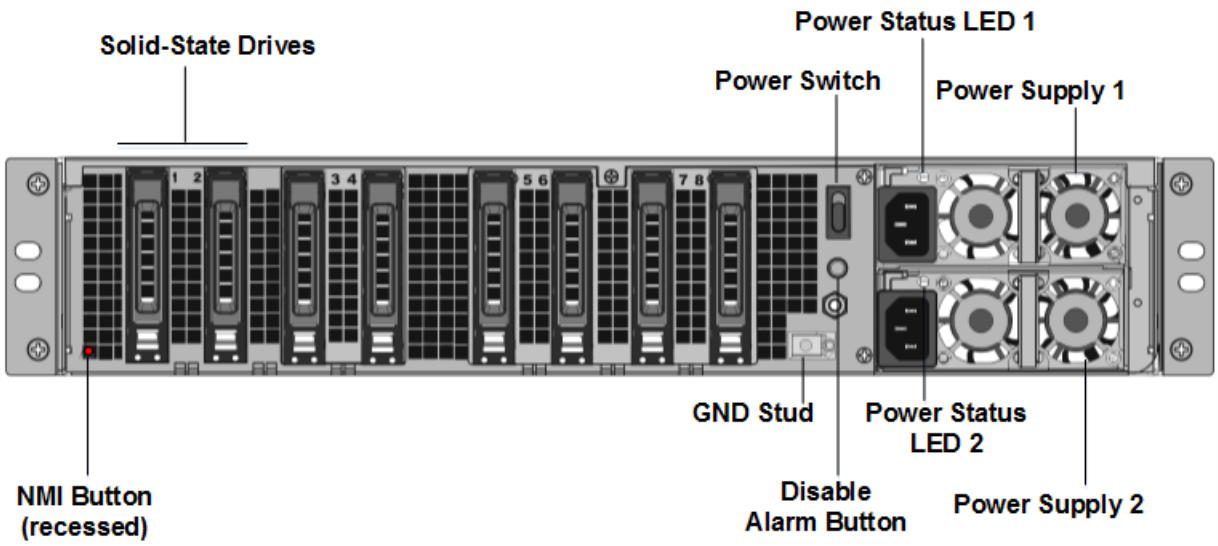
  Note the following points regarding the network ports on 14000-40G appliances:

  * 10G ports do not support 1G copper or 1G fiber transceivers.

  * 40G ports do not support 10G and 1G transceivers.

The following figure shows the back panel of the MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G appliance.

Figure 2. Citrix NetScaler MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, back panel



The following components are visible on the back panel of the MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G

appliance:

- Two 240 GB removable solid-state drives (SSDs).

  These appliances are redundant array of independent disks (RAID) devices. In a RAID configuration, the same data is stored on multiple drives to improve performance, increase storage capacity, lower the risk of data loss, and provide fault tolerance.

  The two SSDs store the same data. If one fails and you replace it, the new SSD mirrors the other one.

- Power switch, which turns power to the appliance on or off. Press the switch for lessthan two seconds to turn off the power.
- Two power supplies, each rated at 1000 watts, 100-240 volts. Each power supply has an LED that indicates the status of the power supply, as described in http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-common-components-ref.html.
- Disable alarm button, which is functional only when the appliance has two powersupplies.
- Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet, or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.
- Non-maskable interrupt (NMI) Button, used at the request of Technical Support toinitiate a core dump. To press this red button, which is recessed to preventunintentional activation, use a pen, pencil, or other pointed object. The NMI Buttonis also available remotely over the network in the LOM GUI, in the Remote Controlmenu. For more information about the lights out management port of the appliance, see http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-lom-intro-wrapper-con.html.

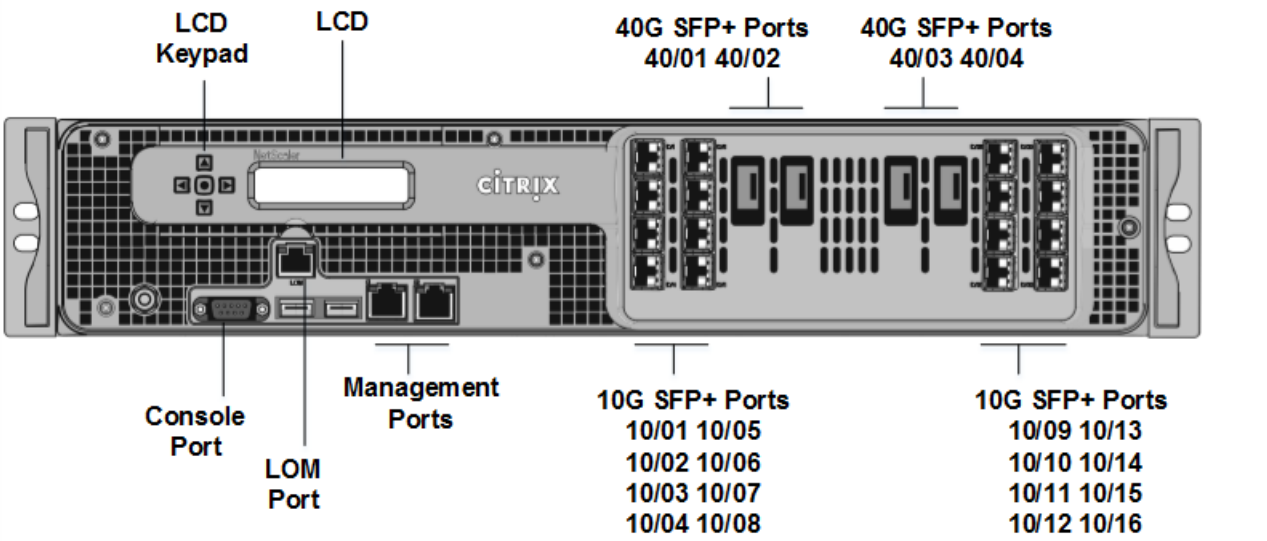
Citrix NetScaler MPX 25100 40G, MPX 25160 40G

Mar 24, 2016

The Citrix NetScaler MPX 25100 40G, MPX 25160 40G are 2U appliances. Each model has two 10-core processors, 256 gigabytes (GB) of memory, four 40G QSFP+ ports, and sixteen 10G SFP+ ports (4x40G QSFP+ + 16x10G SFP+).

The following figure shows the front panel of the MPX 25100 40G, MPX 25160 40G (4x40G QSFP+ + 16x10G SFP+) appliance.

Figure 1. Citrix NetScaler MPX 25100 40G, MPX 25160 40G (4x40G QSFP+ + 16x10G SFP+), front panel



The NetScaler MPX 25100 40G, MPX 25160 40G appliances have the following ports:

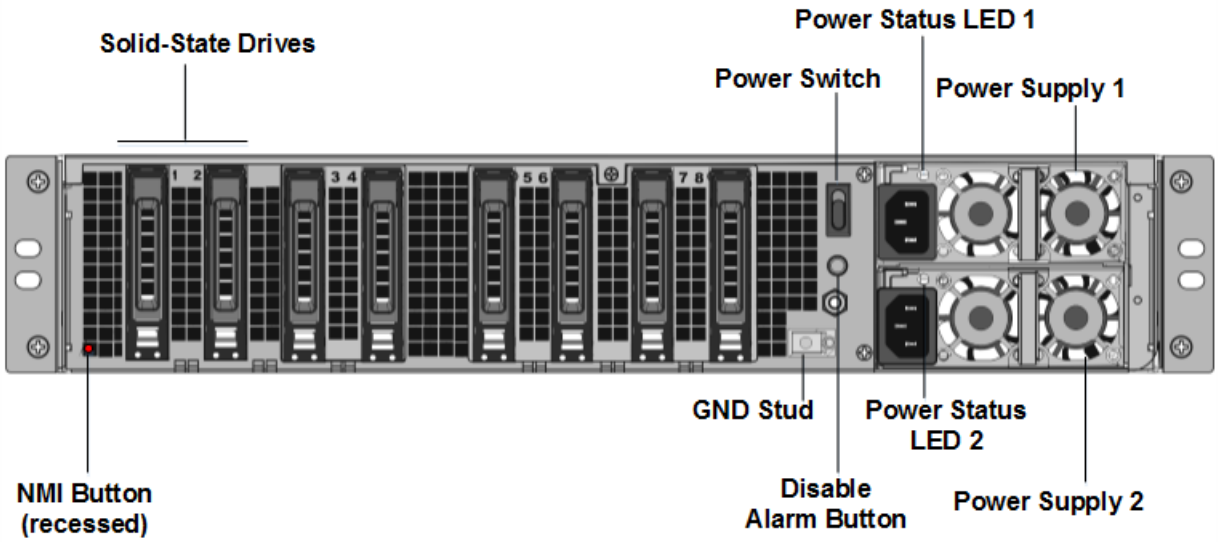
- RS232 serial Console Port.
- 10/100Base-T copper Ethernet Port (RJ45), also called the LOM port. You can use this port to remotely monitor and manage the appliance independently of the NetScaler software.
- Two 10/100/1000Base-T copper Ethernet Management Ports (RJ45), numbered 0/1 and 0/2 from left to right. These ports are used to connect directly to the appliance for system administration functions.
- Network Ports, four 40G QSFP+ ports and sixteen 10G SFP+ ports (4x40G QSFP+ + 16x10G SFP+).

  Note the following points regarding the network ports on 25000-40G appliances:

  - 10G ports do not support 1G copper or 1G fiber transceivers.
  - 40G ports do not support 10G and 1G transceivers.

The following figure shows the back panel of the MPX 25100 40G, MPX 25160 40G appliance.

Figure 2. Citrix NetScaler MPX 25100 40G, MPX 25160 40G, back panel



The following components are visible on the back panel of the MPX 25100 40G, MPX 25160 40G

appliance:

- Two 300 GB removable solid-state drives.

  These appliances are redundant array of independent disks (RAID) devices. In a RAID configuration, the same data is stored on multiple drives to improve performance, increase storage capacity, lower the risk of data loss, and provide fault tolerance.

- Power switch, which turns power to the appliance on or off. Press the switch for less than two seconds to turn off the power.
- Two power supplies, each rated at 1000 watts, 100-240 volts. Each power supply has an LED that indicates the status of the power supply, as described in http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-common-components-ref.html.
- Disable alarm button, which is functional only when the appliance has two powersupplies.

  Press this button to stop the power alarm from sounding when you have plugged the appliance into only one power outlet, or when one power supply is malfunctioning and you want to continue operating the appliance until it is repaired.

- Non-maskable interrupt (NMI) Button, used at the request of Technical Support toinitiate a core dump. To press this red button, which is recessed to preventunintentional activation, use a pen, pencil, or other pointed object. The NMI Buttonis also available remotely over the network in the LOM GUI, in the Remote Control menu. For more information about lights out management port of the appliance, see http://docs.citrix.com/en-us/netscaler/10-1/ns-gen-hardware-wrapper-10-con/ns-hardware-lom-intro-wrapper-con.html.

Application Firewall Platforms

Oct 23, 2013

For information about the Application Firewall platforms, see Citrix NetScaler Application Firewall.

Summary of Hardware Specifications

Apr 01, 2016

The following tables summarize the specifications of the hardware platforms.

Table 1. MPX Platform Summary

	MPX 5500	MPX 5550/MPX 5650	MPX 7500/MPX 9500	MPX 15000
Processors	1 dual-core	1 quad-core	1 dual-core	2 quad-core
Memory	4 GB	8 GB	8 GB	16 GB
Ports - 1G	4x10/100/1000Base-T copper Ethernet ports	6x10/100/1000Base-T copper Ethernet ports	8x10/100/1000Base-T copper Ethernet ports model: 8x10/100/1000Base-T copper Ethernet ports 4x1G SFP + 4x10/100/1000Base-T copper Ethernet ports model: 4xcopper/fiber 1G SFP ports, 4x10/100/1000Base-T copper Ethernet ports	8x10/100/1000Base-T copper Ethernet ports
Ports - 10G	NA	NA	NA	NA
Number of Power Supplies	1	1	1 with second optional	2
AC Power Supply input voltage, frequency, & current	100–240 VAC 50–60 Hz 3–1.5 A	100–240 VAC 50–60 Hz 2.5 A	100–240 VAC 50–60 Hz 3–1.5 A	100–240 VAC 47–63 Hz
Maximum Power Consumption	260 W	300 W	260 W	700 W

Heat Dissipation	887 BTU per hour	630 BTU per hour	887 BTU per hour	
Weight	22 lbs 9.98 kg	32 lbs 14.5 kg	23 lbs with one power supply 10.43 kg with one power supply	52 lbs 23.58 kg
Height	1U	1U	1U	2U
Width	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks
Depth	21.75 in or 55 cm	24.02 in or 61 cm	21.75 in or 55 cm	18.5 in or 47 cm
Operating Temperature	0–40° C 32–104° F	0–40° C 32–104° F	0–40° C 32–104° F	0–35° C 32–95° F
Humidity range (non-condensing)	5%–95%	5%–95%	5%–95%	5%–95%
Safety Certifications	CSA	CSA	CSA	UL & TUV-C
EMC & Susceptibility	FCC (Part 15 Class A), CE, C-Tick, CCC, KCC, NOM, PCT, VCCI, SASO, SABS	FCC (Part 15 Class A), CE, C-Tick, VCCI-A, CCC, KCC, NOM, SASO, SABS, PCT	FCC (Part 15 Class A), CE, C-Tick, CCC, KCC, NOM, PCT, VCCI, SASO, SABS	FCC (Part 15 Class A), DoC, CE, VCCI, CNS, AN/NES
Compliance	RoHS, WEEE	RoHS, WEEE	RoHS, WEEE	RoHS, WEEE

Table 2. MPX Platform Summary (contd.)

	MPX 17000	MPX 8005/MPX 8015/MPX 8200/MPX 8400/MPX 8600/MPX 8800	MPX 9700/MPX 10500/MPX 12500/MPX 15500	MPX 11500/MPX 13500/MPX 14500/MPX 16500/MPX 18500/ MPX 20500
Processors	2 quad-core	1 quad-core	2 quad-core	2 six-core
Memory	32 GB	32 GB	16 GB	48 GB
Ports - 1G	Ten network-port	6x1G SFP +	8x10/100/1000Base-T	8x1G SFP ports

	model: **MPX 17000** 8x10/100/1000Base-T copper Ethernet ports	**6x10/100/1000Base-T copper Ethernet model:** **MPX 8005/MPX 8015/MPX 8200/MPX 8400/MPX 8600/MPX 8800** 6xcopper/fiber 1G SFP ports, 6x10/100/1000Base-T copper Ethernet ports **2x10G SFP+ 6x10/100/1000Base-T copper Ethernet model:** 6xcopper/fiber 1G SFP ports	copper Ethernet ports, 8xcopper/fiber 1G SFP ports **MPX 9700/MPX 10500/MPX 12500/MPX 15500** **10G and FIPS model:** 8xcopper/fiber 1G SFP ports	**MPX 11500/MPX 13500/MPX 14500/MPX 16500/MPX 18500/ MPX 20500**
Ports - 10G	**Four network-port model:** 4x10G XFP ports **Ten network-port model:** 2x10G XFP ports	**2x10G SFP+ 6x10/100/1000Base-T copper Ethernet model:** 2x10G SFP+ Ports	**10G and FIPS model:** 2x10G SFP+ ports	4x10G SFP+ ports
Number of Power Supplies	2	1	2	2
AC Power Supply input voltage, frequency, & current	100–240 VAC 47–63 Hz	100–240 VAC 50–60 Hz 2.5 A	100–240 VAC 50–60 Hz 4.5–2.5 A	100–240 VAC 50–60 Hz 6.5–3.5 A
Maximum Power Consumption	700 W	450 W	450 W	650 W
Heat Dissipation		630 BTU per hour	1550 BTU per hour	2200 BTU per hour
Weight	52 lbs 23.59 kg	32 lbs 14.52 kg	31 lbs 14.06 kg	46 lbs 20.87 kg
Height	2U	1U	2U	2U
Width	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks

	MPX 17000	MPX 8005/MPX 8015/MPX 8200/MPX 8400/MPX 8600/MPX 8800	MPX 9700/MPX 10500/MPX 12500/MPX 15500	MPX 11500/MPX 13500/MPX 14500/MPX 16500/MPX 18500/ MPX 20500
Depth	18.5 in or 47 cm	24.01 in or 61 cm	24.5 in or 62 cm	28 in or 71.68 cm
Operating Temperature	0–35° C 32–95° F	0–40° C 32–104° F	0–40° C 32–104° F	0–40° C 32–104° F
Humidity range (non-condensing)	5%–95%	5%–95%	5%–95%	5%–95%
Safety Certifications	UL & TUV-C	TUV	CSA	CSA
EMC & Susceptibility	FCC (Part 15 Class A), DoC, CE, VCCI, CNS, AN/NES	FCC (Part 15 Class A), CE, C-Tick, VCCI-A	FCC (Part 15 Class A), CE, C-Tick, KCC, NOM, PCT, VCCI, SASO, SABS	FCC (Part 15 Class A), CE, C-Tick, VCCI, CCC, KC, NOM, GOST, SABS, SASO
Compliance	RoHS, WEEE	RoHS, WEEE	RoHS, WEEE	RoHS, SVHC, WEEE

Table 3. MPX Platform Summary (contd.)

	MPX 11515/11520/11530/11540/11542	MPX 17500/MPX 19500/MPX 21500	MPX 17550/MPX 19550/MPX 20550/MPX 21550	MPX 22040/MPX 22060/MPX 22080/MPX 22100/MPX 22120	MPX 24100/MPX 24150
Processors	2 six-core	2 six-core	2 six-core	2 eight-core	2 eight-core
Memory	48 GB	48 GB	96 GB	256 GB	256 GB
Ports - 1G	4xcopper/fiber 1G SFP ports	NA	NA	**12x1G SFP + 24x10G SFP+ model**: 12xcopper/fiber 1G SFP ports	**12x1G SFP + 24x10G SFP+ model**: 12xcopper/fiber 1G SFP ports
Ports - 10G	8x10G SFP+ ports	8x10G SFP+ ports	8x10G SFP+ ports	**12x1G SFP + 24x10G SFP+ model**: 24x10G SFP+ ports **24x10G SFP+**	**12x1G SFP + 24x10G SFP+ model**: 24x10G SFP+ ports

	MPX 11515/11520/11530/11540/11542	MPX 17500/MPX 19500/MPX 21500	MPX 17550/MPX 19550/MPX 20550/MPX 21550	MPX 22040/MPX 22060/MPX 22080/MPX 22100/MPX 22120	MPX 24100/MPX 24150
Number of Power Supplies	2	2	2	4	4
AC Power Supply input voltage, frequency, & current	100–240 VAC 50–60 Hz 6.5–3.5 A	100–240 VAC 50–60 Hz 6.5–3.5 A	100–240 VAC 50–60 Hz 6.5–3.5 A	**12x1G SFP + 24x10G SFP+ model:** 100-240VAC 50/60Hz 6.0-12.0A **24x10G SFP+ model:** 100-240VAC 50/60Hz 6.5-15.5A	**12x1G SFP + 24x10G SFP+ model:** 100-240VAC 50/60Hz 6.0-12.0A
Maximum Power Consumption	650 W	650 W	850 W	**12x1G SFP + 24x10G SFP+ model:** 1050 W **24x10G SFP+ model:** 1400 W	**12x1G SFP + 24x10G SFP+ model:** 1050 W
Heat Dissipation	2200 BTU per hour	2200 BTU per hour	2900 BTU per hour	**12x1G SFP + 24x10G SFP+ model:** 2,000-2,6000 BTU per hour **24x10G SFP+ model:** 2,700-3,800 BTU per hour	**12x1G SFP + 24x10G SFP+ model:** 2,000-2,6000 BTU per hour
Weight	46 lbs	40 lbs	40 lbs	85 lbs	85 lbs

	20.87 kg	18.14 kg	18.14 kg	38.56 kg	38.56 kg
Height Width	MPX 11515/11520/11530/11540/11542 2U EIA 310-D for 19-inch racks	MPX 17500/MPX 19500/MPX 21500 2U EIA 310-D for 19-inch racks	MPX 17550/MPX 19550/MPX 20550/MPX 21550 2U EIA 310-D for 19-inch racks	MPX 22040/MPX 22060/MPX 22080/MPX 22100/MPX 22120 2U EIA 310-D for 19-inch racks	MPX 24100/MPX 24150 2U EIA 310-D for 19-inch racks
Depth	28 in or 71.68 cm	24.75 in or 62.865 cm	24.75 in or 62.865 cm	28¼ in or 72 cm	28¼ in or 72 cm
Operating Temperature	0–40° C 32–104° F	0–40° C 32–104° F	0–40° C 32–104° F	0–40° C 32–104° F	0–40° C 32–104° F
Humidity range (non-condensing)	5%–95%	5%–95%	5%–95%	20%–80%	20%–80%
Safety Certifications	CSA	TUV	TUV	CSA	CSA
EMC & Susceptibility	FCC (Part 15 Class A), CE, C-Tick, VCCI, CCC, KC, NOM, GOST, SABS, SASO	FCC (Part 15 Class A), CE, C-Tick, VCCI-A	FCC (Part 15 Class A), CE, C-Tick, VCCI-A	FCC (Part 15 Class A), CE (EN55022/55024), C-Tick, VCCI	FCC (Part 15 Class A), CE (EN55022/55024), C-Tick, VCCI
Compliance	RoHS, SVHC, WEEE	RoHS, WEEE	RoHS, WEEE	RoHS, WEEE	RoHS, WEEE

Table 4. MPX Platform Summary (contd.)

	MPX 25100T/MPX 25160T	MPX 14020/14030/14040/14060/14080/14100	MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G	MPX 25100 40G, MPX 25160 40G
Regulatory model number	Citrix 2U1P1X	Citrix 2U1P1B	Citrix 2U1P1B	Citrix 2U1P1D
Processor	2 ten-core	2 six-core	2 six-core	2 ten-core

Memory	128 GB	64 GB	64 GB	256 GB
Number of Power Supplies	2	2	2	2
AC power supply input voltage, frequency, and current	100-240V AC 50-60 Hz 8.0 – 4.0 A	100-240V AC 50-60 Hz 5.9 – 2.95 A	100-240V AC, 50-60 Hz 7.0– 3.5A	100 - 240V AC, 50-60 Hz 9.0 – 4.5A
DC power supply input voltage and current	-36V to -72V DC 22.4 – 11.2A	-36V to -72V DC 16.5 – 8.25 A	-36V to -72V DC, 20.0 – 10.0 A	-36V to -72V DC, 25.5 – 13.0A
Maximum AC power Consumption	717 W	528 W	633 W	822 W
Maximum DC power Consumption	806 W	594 W	712 W	925 W
Airflow (front to rear)	110 CFM Typical	110 CFM Typical 175 CFM Maximum	110 CFM Typical 175 CFM Maximum	110 CFM Typical 175 CFM Maximum
Heat dissipation	2027 BTU per hour Typical	1412 BTU per hour Typical	1412 BTU per hour Typical	1980 BTU per hour Typical
Weight (lbs.)	60	60	60	60
System Weight (lbs.)	39	39	39	39
Height	2U	2U	2U	2U
Width	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks	EIA 310-D for 19-inch racks
Depth	72 cm; 28¼ in	72 cm; 28¼ in	71.2 cm; 28 in	71.2 cm; 28 in
Operating	0-40°C; 32-	0-40°C; 32-104°F	0-40°C; 32-	0-40°C; 32-

Temperature	104°F		104°F	104°F
Humidity range (non-condensing)	20%–80%	20%-80%	5%-95%	5%-95%
Safety Certifications	CSA	CSA	CSA	CSA
EMC & Susceptibility	USA (FCC), Europe (CE), Japan (VCCI), Australia (RCM), China (CCC), Korea (KCC), India (BIS), Mexico (NOM), Saudi Arabia (CITC), South Africa (ICASA), Russia (EAC, CU-TR), Taiwan (BSMI), Brazil (Inmetro & Anatel), Israel (MoE, MoC)	USA (FCC), Europe (CE), Japan (VCCI), Australia (RCM), China (CCC), Korea (KCC), India (BIS), Mexico (NOM), Saudi Arabia (CITC), South Africa (ICASA), Russia (EAC, CU-TR), Taiwan (BSMI), Brazil (Inmetro & Anatel), Israel (MoE, MoC)	USA (FCC), Europe (CE), Japan (VCCI), Australia (RCM), China (CCC), Korea (KCC), India (BIS), Mexico (NOM), Saudi Arabia (CITC), South Africa (ICASA), Russia (EAC, CU-TR), Taiwan (BSMI), Brazil (Inmetro & Anatel), Israel (MoE, MoC)	USA (FCC), Europe (CE), Japan (VCCI), Australia (RCM), China (CCC), Korea (KCC), India (BIS), Mexico (NOM), Saudi Arabia (CITC), South Africa (ICASA), Russia (EAC, CU-TR), Taiwan (BSMI), Brazil (Inmetro & Anatel), Israel (MoE, MoC)
Compliance	RoHS, WEEE, REACH	WEEE, RoHS, REACH	WEEE, RoHS, REACH	WEEE, RoHS, REACH

Hardware Health Attributes

Jan 21, 2014

Operating ranges for NetScaler hardware platforms vary for different attributes. You can use the stat system -detail command to display the current values of the attributes.

You can also query SNMP OIDs to monitor the attributes of the appliance. For more information about SNMP OIDs, see "SNMP OID Reference."

The following table lists the health attributes and their recommended value ranges.

	SNMP Alarm Support	Recommended Range								
		MPX 5500/5600	MPX 7500/9500	MPX 9700/10500/ 12500/15500	MPX 9700/10500/ 12500/15500 10G	MPX 17500/19500/ 21500	MPX 11500/13500/ 14500/16500/ 18500/20500	MPX 17550/19550/ 20550/21550	MPX 8005/8015/8200/8400/ 8600/8800	MPX 5550/5650/ 5750
CPU 0 core (Volts)	No	0.97–1.5	1–1.5	1–1.5	1–1.5	0.99–1.5	0.95–1.5	0.95–1.5	- NA -	- NA -
CPU 1 core (Volts)	No	0.97–1.5	1–1.5	1–1.5	1–1.5	0.99–1.5	0.95–1.56	0.95–1.5	- NA -	- NA -
Main 3.3 V Supply (Volts)	Yes	3.2–3.6	3.2–3.54	3.2–3.54	3.2–3.55	3.19–3.55	3.19–3.55	3.18–3.55	3.14–3.47	3.14–3.47
Standby 3.3 V Supply (Volts)	Yes	3.2–3.6	3.2–3.54	3.2–3.54	3.2–3.55	3.2–3.55	3.1–3.55	3.1–3.55	3.14–3.47	3.14–3.47
+5.0 V Supply (Volts)	No	4.8–5.2	4.8–5.2	4.8–5.2	4.8–5.2	4.8–5.2	4.8–6.24	4.8–5.2	4.75–5.25	4.75–5.25
+12.0 V Supply (Volts)	No	11.5–12.35	11.52–12.35	11.5–12.31	11.8–12.35	11.5–12.35	11.8–12.35	11.5–12.35	11.40–12.60	11.40–12.60
-12.0 V Supply (Volts)	No	- NA -	- NA -	- NA -	- NA -	- NA -	- NA -	- NA -	(-10.80)–(-13.20)	(-10.80)–(-13.20)
Battery (Volts)	No	3–3.5	2.85–3.5	2.85–3.5	2.85–3.5	2.85–3.37	3–3.5	2.8–3.5	> 2.5	> 2.5
Intel CPU Vtt Power (Volts)	No	1–1.2	1–1.2	1–1.2	1–1.2	1–1.2	1–1.2	1–1.2	- NA -	- NA -
5V Standby (Volts)	No	4.9–5.2	4.9–5.2	4.9–5.2	4.9–5.2	4.88–5.2	4.8–5.25	4.9–5.3	- NA -	- NA -
Voltage Sensor2(Volts)	No	1.2–2	1.2–2	1.2–2	1–1.8	1.4–5.2	1.4–6.24	1.4–5.2	3.14–3.47	3.14–3.47
CPU Fan 0 Speed (RPM)	Yes	3000–16000	3000–16000	3000–10000	3000–16000	3000–16000	3000–16000	3000–16000	> 5500	> 5500
CPU Fan 1 Speed (RPM)	Yes	3000–16000	3000–16000	3000–16000	3000–16000	3000–16000	3000–16000	3000–16000	> 5500	> 5500
System Fan Speed (RPM)	Yes	900–15000	900–13000	900–10000	900–9000	900–15000	900–15000	900–15000	> 5500	> 5500
System Fan 1 Speed (RPM)	No	900–15000	900–15000	900–10000	900–8000	900–15000	900–15000	900–16000	> 5500	> 5500
System Fan 2 Speed (RPM)	No	900–15000	900–15000	900–10000	900–10000	900–15000	900–15000	900–16000	> 5500	> 5500
CPU 0	Yes	24–90° C	24–90° C	24–90° C	24–90° C	24–90° C	24–90° C	24–90° C	< 85° C	< 85° C

Temperature		75.2–194° F	75.2–194° F	75.2–194° F	75.2–194° F	75.2–194° F	75.2–194° F	75.2–194° F	< 185° F	< 185° F
CPU 1 Temperature	Yes	24–90° C 75.2–194° F	24–90° C 75.2–194° F	24–90° C 75.2–194° F	24–90° C 75.2–194° F	24–90° C 75.2–194° F	24–90° C 75.2–194° F	24–90° C 75.2–194° F	- NA -	- NA -
Internal Temperature	Yes	19–50° C 66.2–122° F	19–50° C 66.2–122° F	19–50° C 66.2–122° F	19–50° C 66.2–122° F	19–50° C 66.2–122° F	19–50° C 66.2–122° F	19–50° C 66.2–122° F	< 55° C < 131° F	< 55° C < 131° F
Power Supply 1 Status	Yes	Not supported	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Power Supply 2 Status	Yes	Not supported	Normal	Normal	Normal	Normal	Normal	Normal	Normal, if both power supplies are installed	Normal, if both power supplies are installed

The following table lists the health attributes for MPX 22040/22060/22080/22100/22120.

		Lower Non Recoverable	Lower Critical	Lower Non Critical	Upper Non Critical	Upper Critical	Upper Non Recoverable
CPU1 Temp	degrees C	0.000	0.000	0.000	90.000	93.000	95.000
CPU2 Temp	degrees C	0.000	0.000	0.000	90.000	93.000	95.000
System Temp	degrees C	-9.000	-7.000	-5.000	80.000	85.000	90.000
Peripheral Temp	degrees C	-9.000	-7.000	-5.000	80.000	85.000	90.000
PCH Temp	degrees C	-11.000	-8.000	-5.000	90.000	95.000	100.000
FPC_Temp 1	degrees C	na	na	na	66.000	70.000	75.000
FPC_Temp 2	degrees C	na	na	na	72.000	76.000	82.000
FPC_Temp 3	degrees C	na	na	na	72.000	76.000	82.000
HDDBP_Temp 1	degrees C	na	na	na	72.000	76.000	82.000
HDDBP_Temp 2	degrees C	na	na	na	72.000	76.000	82.000
FAN 1	RPM	na	1980.000	na	na	na	na
FAN 2	RPM	na	1980.000	na	na	na	na
FAN 3	RPM	na	1980.000	na	na	na	na
FAN 4	RPM	na	1980.000	na	na	na	na
FAN 5	RPM	na	1980.000	na	na	na	na
FAN 6	RPM	na	1980.000	na	na	na	na
FAN 7	RPM	na	1980.000	na	na	na	na
FAN 8	RPM	na	1980.000	na	na	na	na
PS_1 Status	discrete	na	na	na	na	na	na

PS_1 FAN	RPM	na	na	na	na	na	na
PS_1 Temp	degrees C	na	na	na	na	na	na
PS_2 Status	discrete	na	na	na	na	na	na
PS_2 FAN	RPM	na	na	na	na	na	na
PS_2 Temp	degrees C	na	na	na	na	na	na
PS_3 Status	discrete	na	na	na	na	na	na
PS_3 FAN	RPM	na	1980.000	na	na	na	na
PS_3 Temp	degrees C	na	na	na	72.000	76.000	82.000
PS_4 Status	discrete	na	na	na	na	na	na
PS_4 FAN	RPM	na	1980.000	na	na	na	na
PS_4 Temp	degrees C	na	na	na	72.000	76.000	82.000
FPC Status	discrete	na	na	na	na	na	na
VTT	Volts	0.816	0.864	0.912	1.344	1.392	1.440
CPU1 Vcore	Volts	0.480	0.512	0.544	1.488	1.520	1.552
CPU2 Vcore	Volts	0.480	0.512	0.544	1.488	1.520	1.552
VDIMM AB	Volts	1.104	1.152	1.200	1.648	1.696	1.744
VDIMM CD	Volts	1.104	1.152	1.200	1.648	1.696	1.744
VDIMM EF	Volts	1.104	1.152	1.200	1.648	1.696	1.744
VDIMM GH	Volts	1.104	1.152	1.200	1.648	1.696	1.744
+1.5V	Volts	1.248	1.296	1.344	1.648	1.696	1.744
3.3V	Volts	2.640	2.784	2.928	3.648	3.792	3.936
+3.3VSB	Volts	2.640	2.784	2.928	3.648	3.792	3.936
5V	Volts	4.096	4.288	4.480	5.504	5.696	6.912
12V	Volts	10.176	10.494	10.812	13.250	13.568	13.886
VBAT	Volts	2.400	2.544	2.688	3.312	3.456	3.600

The following table lists the health attributes for MPX 24100/24150.

		Lower Non Recoverable	**Lower Critical**	**Lower Non Critical**	**Upper Non Critical**	**Upper Critical**	**Upper Non Recoverable**
CPU1 Temp	degrees C	0.000	0.000	0.000	90.000	93.000	95.000
CPU2 Temp	degrees C	0.000	0.000	0.000	90.000	93.000	95.000
System Temp	degrees C	-9.000	-7.000	-5.000	80.000	85.000	90.000
Peripheral Temp	degrees C	-9.000	-7.000	-5.000	80.000	85.000	90.000

PCH Temp	degrees C	-11.000	-8.000	-5.000	90.000	95.000	100.000
FPC_Temp 1	degrees C	na	na	na	66.000	70.000	75.000
FPC_Temp 2	degrees C	na	na	na	72.000	76.000	82.000
FPC_Temp 3	degrees C	na	na	na	72.000	76.000	82.000
HDDBP_Temp 1	degrees C	na	na	na	72.000	76.000	82.000
HDDBP_Temp 2	degrees C	na	na	na	72.000	76.000	82.000
FAN 1	RPM	na	1980.000	na	na	na	na
FAN 2	RPM	na	1980.000	na	na	na	na
FAN 3	RPM	na	1980.000	na	na	na	na
FAN 4	RPM	na	1980.000	na	na	na	na
FAN 5	RPM	na	1980.000	na	na	na	na
FAN 6	RPM	na	1980.000	na	na	na	na
FAN 7	RPM	na	1980.000	na	na	na	na
FAN 8	RPM	na	1980.000	na	na	na	na
PS_1 Status	discrete	na	na	na	na	na	na
PS_1 FAN	RPM	na	1980.000	na	na	na	na
PS_1 Temp	degrees C	na	na	na	72.000	76.000	82.000
PS_2 Status	discrete	na	na	na	na	na	na
PS_2 FAN	RPM	na	na	na	na	na	na
PS_2 Temp	degrees C	na	na	na	na	na	na
PS_3 Status	discrete	na	na	na	na	na	na
PS_3 FAN	RPM	na	na	na	na	na	na
PS_3 Temp	degrees C	na	na	na	na	na	na
PS_4 Status	discrete	na	na	na	na	na	na
PS_4 FAN	RPM	na	na	na	na	na	na
PS_4 Temp	degrees C	na	na	na	na	na	na
FPC Status	discrete	na	na	na	na	na	na
VTT	Volts	0.816	0.864	0.912	1.344	1.392	1.440
CPU1 Vcore	Volts	0.480	0.512	0.544	1.488	1.520	1.552
CPU2 Vcore	Volts	0.480	0.512	0.544	1.488	1.520	1.552

VDIMM AB	Volts	1.104	1.152	1.200	1.648	1.696	1.744
VDIMM CD	Volts	1.104	1.152	1.200	1.648	1.696	1.744
VDIMM EF	Volts	1.104	1.152	1.200	1.648	1.696	1.744
VDIMM GH	Volts	1.104	1.152	1.200	1.648	1.696	1.744
+1.5V	Volts	1.248	1.296	1.344	1.648	1.696	1.744
3.3V	Volts	2.640	2.784	2.928	3.648	3.792	3.936
+3.3VSB	Volts	2.640	2.784	2.928	3.648	3.792	3.936
5V	Volts	4.096	4.288	4.480	5.504	5.696	6.912
12V	Volts	10.176	10.494	10.812	13.250	13.568	13.886
VBAT	Volts	2.400	2.544	2.688	3.312	3.456	3.600

Preparing for Installation

Apr 04, 2016

Before you install your new appliance, carefully unpack your appliance and make sure that all parts were delivered. Once you are satisfied that your appliance has been delivered to your expectations, verify that the location where the appliance will be installed meets temperature and power requirements and that the server cabinet or floor-to-ceiling cabinet is securely bolted to the floor and has sufficient airflow.

Only trained and qualified personnel should install, maintain, or replace the appliance, and efforts should be taken to ensure that all cautions and warnings are followed.

This document includes the following information:

The hardware accessories for your particular appliance, such as cables, adapters, and rail kit, vary depending on the hardware platform you ordered. Unpack the box that contains your new appliance on a sturdy table with plenty of space and inspect the contents.

Use the following list to verify that you received everything that should have been included in the box.

- The appliance you ordered
- One RJ-45 to DB-9 adapter
- One 6 ft RJ-45/DB-9 cable
- The following list specifies the number of power cables included for each appliance model:
  - One power cable for the MPX 5500, MPX 5550/5650, MPX 7500/9500, and MPX 8005/8015/8200/8400/8600/8800 appliances
  - Two power cables for the MPX 15000, MPX 17000, MPX 9700/10500/12500/15500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14000, MPX 17500/19500/21500, MPX 25100T/25160T, MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G appliances
  - Four power cables for the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 appliances
    Note: Make sure that a power outlet is available for each cable.

  Note: For Brazilian customers, Citrix does not ship a power cable. Use a cable that conforms to the **ABNT NBR 14136:2002** standard.
- One standard 4-post rail kit
  Note: If the kit that you received does not fit your rack, contact your Citrix sales representative to order the appropriate kit.

In addition to the items included in the box with your new appliance, you will need the following items to complete the installation and initial configuration process.

- Ethernet cables for each additional Ethernet port that you will connect to your network
- One available Ethernet port on your network switch or hub for each NetScaler Ethernet port you want to connect to your network
  Note: Transceiver modules are sold separately. Contact your Citrix sales representative to order transceiver modules for your appliance. Only transceivers supplied by Citrix are supported on the appliance.
- A computer to serve as a management workstation

There are specific site and rack requirements for the NetScaler appliance. You must make sure that adequate environmental control and power density are available. Racks must be bolted to the ground, have sufficient airflow, and have adequate power and network connections. Preparing the site and rack are important steps in the installation process and help ensure a smooth installation.

Site Requirements

The appliance should be installed in a server room or server cabinet with the following features:

Environment control

An air conditioner, preferably a dedicated computer room air conditioner (CRAC), capable of maintaining the cabinet or server room at a temperature of no more than 27 degrees C/80.6 degrees F at altitudes of up to 2100 m/7000 ft, or 18 degrees C/64.4 degrees F at higher altitudes, a humidity level no greater than 45 percent, and a dust-free environment.

Power density

Wiring capable of handling at least 4,000 watts per rack unit in addition to power needs for the CRAC.

Rack Requirements

The rack on which you install your appliance should meet the following criteria:

Rack characteristics

Racks should be either integrated into a purpose-designed server cabinet or be the floor-to-ceiling type, bolted down at both top and bottom to ensure stability. If you have a cabinet, it should be installed perpendicular to a load-bearing wall for stability and sufficient airflow. If you have a server room, your racks should be installed in rows spaced at least 1 meter/3 feet apart for sufficient airflow. Your rack must allow your IT personnel unfettered access to the front and back of each server and to all power and network connections.

Power connections

At minimum, two standard power outlets per unit.

Network connections

At minimum, four Ethernet connections per rack unit.

Space requirements

One empty rack unit for the Citrix NetScaler MPX 5500, MPX 5550/5650, MPX 7500/9500, and MPX 8005/8015/8200/8400/8600/8800, MPX 14000, and two consecutive empty rack units for all other appliance models.

Note: You can order the following rail kits separately.

- Compact 4-post rail kit, which fits racks of 23 to 33 inches.
- 2-post rail kit, which fits 2-post racks.

Electrical Safety Precautions

Caution: During installation or maintenance procedures, wear a grounding wrist strap to avoid ESD damage to the electronics of the appliance. Use a conductive wrist strap attached to a good earth ground or to the appliance. You can attach it to the connector beside the ESD symbol on the back.
Follow basic electrical safety precautions to protect yourself from harm and the appliance from damage.

- Be aware of the location of the emergency power off (EPO) switch, so that you can quickly remove power to the appliance if an electrical accident occurs.
- Remove all jewelry and other metal objects that might come into contact with power sources or wires before installing or repairing the appliance. When you touch both a live power source or wire and ground, any metal objects can heat up rapidly and may cause burns, set clothing on fire, or fuse the metal object to an exposed terminal.
- Use a regulating, uninterruptible power supply (UPS) to protect the appliance from power surges and voltage spikes, and to keep the appliance operating in case of power failure.
- Never stack the appliance on top of any other server or electronic equipment.
- All appliances are designed to be installed on power systems that use TN earthing. Do not install your device on a power

system that uses either TT or IT earthing.

- Make sure that the appliance has a direct physical connection to the earth during normal use. When installing or repairing an appliance, always make sure that the ground circuit is connected first and disconnected last.
- Make sure that a fuse or circuit breaker no larger than 120 VAC, 15 A U.S. (240 VAC, 16 A international) is used on all current-carrying conductors on the power system to which your appliances are connected.
- Do not work alone when working with high voltage components.
- Always disconnect the appliance from power before removing or installing any component. When disconnecting power, first shut down the appliance, and then unplug the power cords of all the power supply units connected to the appliance. As long as the power cord is plugged in, line voltages can be present in the power supply, even when the power switch is OFF.
- Do not use mats designed to decrease static electrical discharge as protection from electrical shock. Instead, use rubber mats that have been specifically designed as electrical insulators.
- Make sure that the power source can handle the appliance's maximum power consumption rating with no danger of an overload. Always unplug any appliance before performing repairs or upgrades.
- Do not overload the wiring in your server cabinet or on your server room rack.
- During thunderstorms, or anticipated thunderstorms, avoid performing any hardware repairs or upgrades until the danger of lightning has passed.
- When you dispose of an old appliance or any components, follow any local and national laws on disposal of electronic waste.
- To prevent possible explosions, replace expired batteries with the same model or a manufacturer-recommended substitute and follow the manufacturer's instructions for battery replacement.
- Never remove a power supply cover or any sealed part that has the following label:

Hazardous voltage, current, and energy levels are present inside any component that has this label attached. There are no user-serviceable parts inside these components. If you suspect a problem with one of these parts, contact Citrix Technical Support.

Appliance Precautions

- Determine the placement of each component in the rack before you install the rails.
- Install the heaviest appliance first, at the bottom of the rack, and then work upward. Distribute the load on the rack evenly. An unbalanced rack is hazardous.
- Allow the power supply units and hard drives to cool before touching them.
- Install the equipment near an electrical outlet for easy access.
- Mount equipment in a rack with sufficient airflow for safe operation.
- For a closed or multiple-unit rack assembly, the ambient operating temperature of the rack environment might be greater than the ambient temperature of the room. Therefore, consider the lowest and highest operating temperatures of the equipment when making a decision about where to install the appliance in the rack.

Rack Precautions

- Make sure that the leveling jacks on the bottom of the rack are fully extended to the floor, with the full weight of the rack resting on them.
- For a single-rack installation, attach a stabilizer to the rack.
- For a multiple-rack installation, couple (attach) the racks together.
- Always make sure that the rack is stable before extending a component from the rack.

- Extend only one component at a time. Extending two or more simultaneously might cause the rack to become unstable.
- The handles on the left and right of the front panel of the appliance should be used only for extending the appliance out of the rack. Do not use these handles for mounting the appliance on the rack. Use the rack-rail hardware, described later, instead.

Installing the Hardware

Apr 04, 2016

After you have determined that the location where you will install your appliance meets the environmental standards and the server rack is in place according to the instructions, you are ready to install the hardware. After you mount the appliance, you are ready to connect it to the network, to a power source, and to the console terminal that you will use for initial configuration. To complete the installation, you turn on the appliance. Be sure to observe the cautions and warnings listed with the installation instructions.

This document includes the following details:

Most appliances can be installed in standard server racks that conform to EIA-310-D specification. The appliances ship with a set of rails, which you must install before you mount the appliance. The only tools that you need for installing an appliance are a Phillips screwdriver and a flathead screwdriver.

Caution: If you are installing the appliance as the only unit in the rack, mount it at the bottom. If the rack contains other units, make sure that the heaviest unit is at the bottom. If the rack has stabilizing devices available, install them before mounting the appliance.
The following table lists the different hardware platforms and the rack units required for each platform.

Table 1. Height Requirements For Each Platform

Platform	**Number of rack units**
MPX 5500	One rack unit
MPX 5550/5650	One rack unit
MPX 7500/9500	One rack unit
MPX 8005/8015/8200/8400/8600/8800	One rack unit
MPX 9700/10500/12500/15500	Two rack units
MPX 14020/14030/14040/ 14060/14080/14100	Two rack units

MPX 15000, MPX 17000	Two rack units
MPX 11500/13500/14500/16500/18500/20500	Two rack units
MPX 11515/11520/11530/11540/11542	Two rack units
MPX 17500/19500/21500	Two rack units
MPX 17550/19550/20550/21550	Two rack units
MPX 22040/22060/22080/22100/22120	Two rack units
MPX 24100/24150	Two rack units
MPX 25100T/25160T	Two rack units
MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G	Two rack units
MPX 25100 40G, MPX 25160 40G	Two rack units

Each appliance ships with a mounting rail kit that contains two rail assemblies, one for the left side and the other for the right side of the appliance, and screws to attach the rails. An assembly consists of an inner rail and a rack rail. The supplied rail kit is 28 inches long (38 inches extended). Contact your Citrix sales representative to order a 23-inch (33 inches extended) rail kit.
Note: The same rail kit is used for both square-hole and round-hole racks. See "Installing the Rail Assembly to the Rack" for specific instructions for threaded, round-hole racks.
To mount the appliance, you must first install the rails and then install the appliance in the rack.

Perform the following tasks to mount the appliance:
- Remove the inner rails from the rail assembly.
- Attach the inner rails to the appliance.
- Install the rack rails on the rack.
- Install the appliance in the rack.

The appliance is shipped with rack-rail hardware. This hardware consists of two inner rails that you attach to the appliance, one on each side, and a rack-rail assembly that you attach to the rack. The following figure illustrates the steps involved in mounting the Citrix NetScaler appliance to a rack.

To remove the inner rails from the rail assembly

1. Place the rail assembly on a flat surface.

2. Slide out the inner rail toward the front of the assembly.
3. Depress the latch until the inner rail comes all the way out of the rail assembly.
4. Repeat steps 1 through 3 to remove the second inner rail.

To attach the inner rails to the appliance

1. Position the right inner rail behind the handle on the right side of the appliance.
2. Align the holes on the rail with the corresponding holes on the side of the appliance.
3. Attach the rail to the appliance with the provided screws: 4 per side for a 1U appliance and 5 per side for a 2U appliance, as shown in the following figure.
   Figure 1. Attaching inner rails

4. Repeat steps 1 through 3 to install the left inner rail on the other side of the appliance.

To install the rack rails on the rack

1. If you have a round-hole, threaded rack, skip to step 3.
2. Install square nut retainers into the front post and back post of the rack as shown in the following figures. Before inserting a screw, be sure to align the square nut with the correct hole for your 1U or 2U appliance. The three holes are not evenly spaced.
   Figure 2. Installing Retainers into the Front Rack Posts

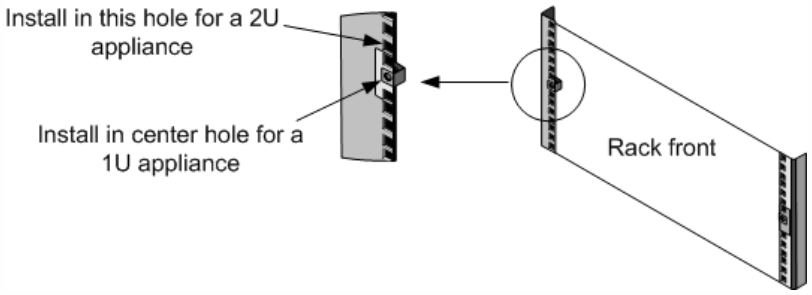
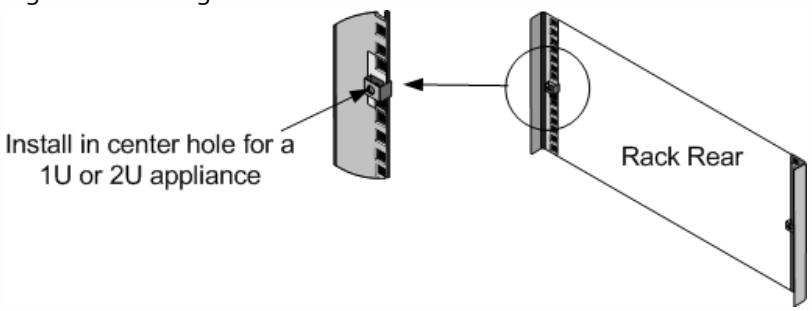
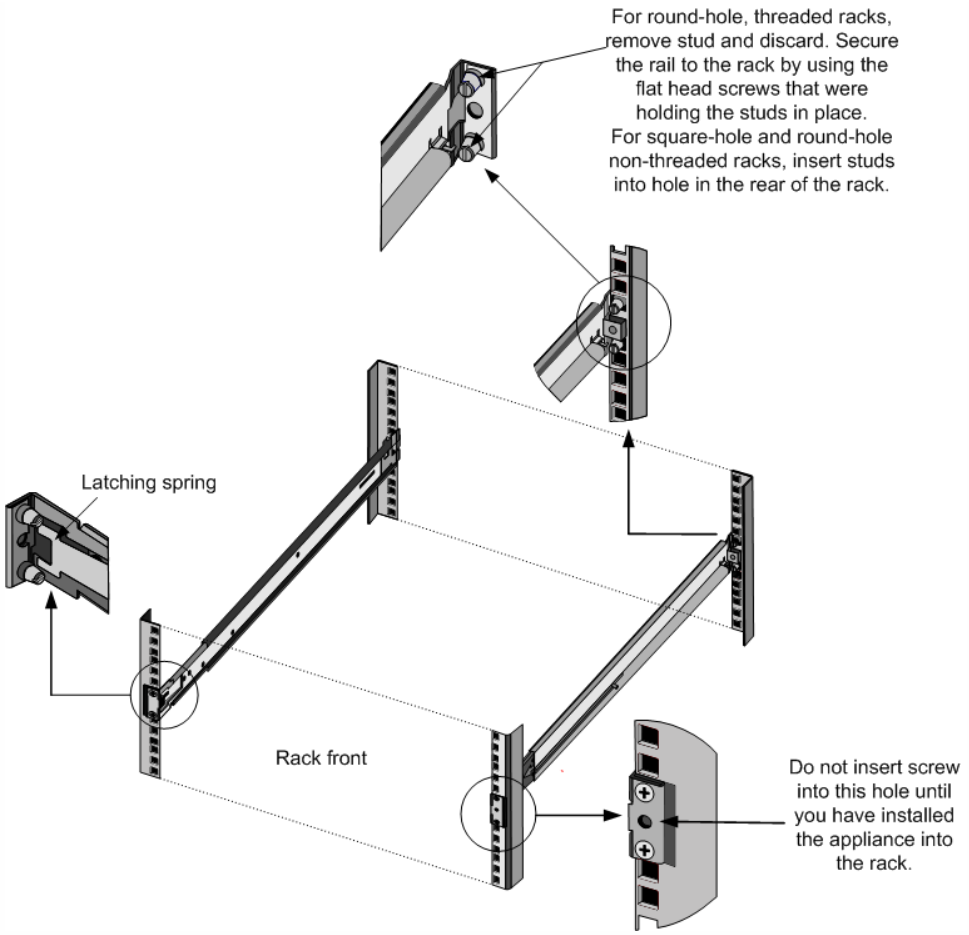


   Figure 3. Installing Retainers into the Rear Rack Posts



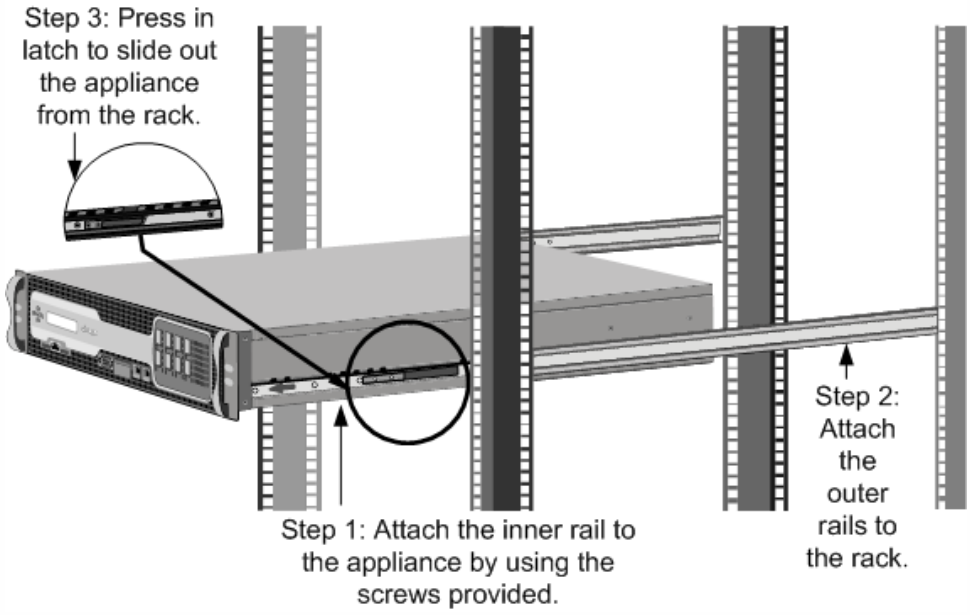
3. Install the adjustable rail assembly into the rack as shown in the following figures. Use a screw to lock the rear rail flange into the rack. With the screw securing the rail in place, you can optionally remove the latching spring.
   Figure 4. Installing the Rail Assembly to the Rack



To install the appliance in the rack

1. Align the inner rails, attached to the appliance, with the rack rails.
2. Slide the appliance into the rack rails, keeping the pressure even on both sides.
3. Verify that the appliance is locked in place by pulling it all the way out from the rack.

Figure 5. Rack Mounting the Appliance



Note: This section applies to the MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 22040/22060/22080/22100/22120, and MPX 24100/24150 appliances.

A Small Form-Factor Pluggable (SFP) is a compact transceiver that can operate at speeds of up to 1 gigabit per second and is available in both copper and fiber types. Inserting a 1G SFP copper transceiver converts the 1G SFP port to a 1000BASE-T port. Inserting a 1G SFP fiber transceiver converts the 1G SFP port to a 1000BASE-X port. Auto-negotiation is enabled by default on the 1G SFP port into which you insert your 1G SFP transceiver. As soon as a link between the port and the network is established, the speed and mode are matched on both ends of the cable.

Note: The 1G SFP transceiver is hot-swappable from release 9.3 build 47.5 and later on the NetScaler appliances that use the e1k interface. The following platforms support 1G SPF transceivers:

- MPX 7500/9500
- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500
- MPX 11500/13500/14500/16500/18500/20500
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150

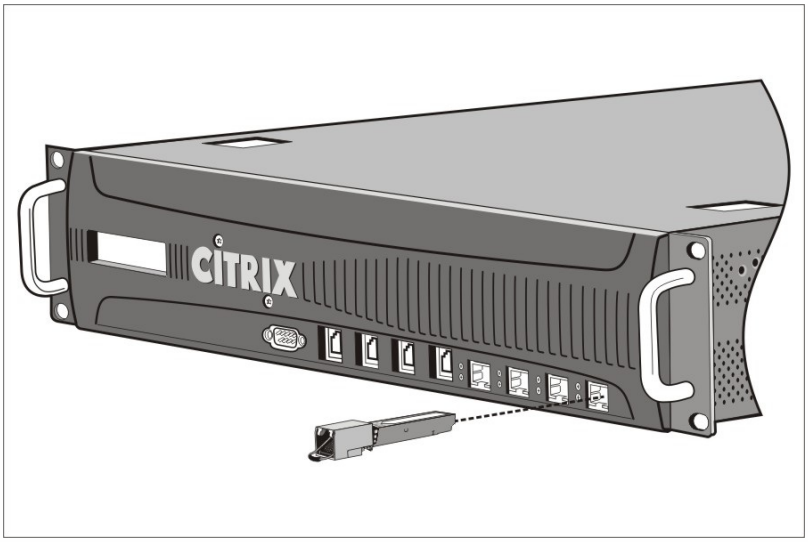
Caution: NetScaler appliances do not support 1G SFP transceivers from vendors other than Citrix Systems. Attempting to install third-party 1G SFP transceivers on your NetScaler appliance voids the warranty.

Insert 1G SFP transceivers into the 1G SFP ports on the front panel of the appliance. Frequent installation and removal of transceivers shortens their life span. Follow the removal procedure carefully to avoid damaging the 1G SFP transceiver or the appliance.

Caution: Do not install the transceivers with the cables attached. Doing so can damage the cable, the connector, or the optical interface of the transceiver.

To install a 1G SFP transceiver

1. Remove the 1G SFP transceiver carefully from its box.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Align the 1G SFP transceiver to the front of the 1G SFP transceiver port on the front panel of the appliance, as shown in the following figure.
   Note: The illustration in the following figures might not represent your actual appliance.
   Figure 6. Installing a 1G SFP transceiver



3. Hold the 1G SFP transceiver between your thumb and index finger and insert it into the 1G SFP transceiver port, pressing it in until you hear the transceiver snap into place.
4. Lock the transceiver.
5. Verify that the LED is green and blinks twice, which indicates that the transceiver is functioning correctly.
6. If you are using a fiber 1G SFP transceiver, do not remove the dust caps attached to the transceiver and the cable until you are ready to insert the cable.

To remove a 1G SFP transceiver

1. Disconnect the cable from the 1G SFP transceiver. If you are using a fiber optic cable, replace the dust cap on the cable before putting it away.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Unlock the 1G SFP transceiver.
3. Hold the 1G SFP transceiver between your thumb and index finger and slowly pull it out of the port.
4. If you are removing a fiber 1G SFP transceiver, replace the dust cap before putting it away.
5. Put the 1G SFP transceiver into its original box or another appropriate container.

Note: This section applies to the MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 15000, MPX 17000, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 14020/14030/14040/14060/14080/14100,MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G, MPX 17500/19500/21500, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, and MPX 25100T/25160T appliances.
A 10-Gigabit Small Form-Factor Pluggable (XFP or SFP+) is a compact optical transceiver that can operate at speeds of up to 10 gigabits per second. The MPX 15000 and MPX 17000 appliances use XFP transceivers and the MPX 8005/8015/8200/8400/8600/8800, MPX 9700/10500/12500/15500, MPX 11500/13500/14500/16500/18500/20500, MPX 14020/14030/14040/14060/14080/14100, MPX 17500/19500/21500, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, MPX 25100T/25160T, MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G, MPX 25100 40G, MPX 25160 40G appliances use 10G SFP+ transceivers. Autonegotiation is enabled by default on the XFP/10G SFP+ ports into which you insert your XFP/10G SFP+ transceiver. As soon as a link between the port and the network is established, the mode is matched on both ends of the cable and for 10G SFP+ transceivers, the speed is also autonegotiated.

Note: An XFP transceiver is **not hot-swappable** on the NetScaler appliances. You must restart a NetScaler appliance after you insert an XFP transceiver.

However, the 10G SFP+ transceiver is hot-swappable from release 9.3 build 57.5 and later on the NetScaler appliances that use the ixgbe (ix) interface. The following platforms support 10G SPF+ transceivers:

- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500 10G and 10G FIPS
- MPX 11500/13500/14500/16500/18500/20500
- MPX 11515/11520/11530/11540/11542

- MPX 14020/14030/14040/14060/14080/14100
- MPX 17500/19500/21500
- MPX 17550/19550/20550/21550
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150
- MPX 25100T/25160T
- MPX 14020 40G, MPX 14040 40G, MPX 14060 40G, MPX 14080 40G
- MPX 25100 40G, MPX 25160 40G

The following platforms support XFP transceivers:

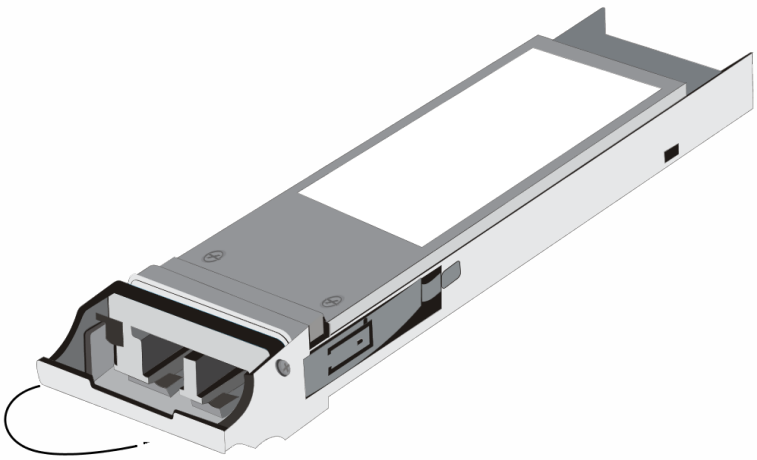
- MPX 15000
- MPX 17000

Caution: NetScaler appliances do not support XFP/10G SFP+ transceivers provided by vendors other than Citrix Systems. Attempting to install third-party XFP/10G SFP+ transceivers on your NetScaler appliance voids the warranty.
Insert the XFP/10G SFP+ transceivers into the XFP/10G SFP+ ports on the front panel of the appliance. Frequent installation and removal of transceivers shortens their life span. Follow the removal procedure carefully to avoid damaging the transceiver or the appliance.

Caution: Do not install the transceivers with the cables attached. Doing so can damage the cable, the connector, or the optical interface of the transceiver.

To install an XFP/10G SFP+ transceiver

1. Remove the XFP/10G SFP+ transceiver carefully from its box.
   Danger: Do not look directly into fiber optic transceivers and cables. They emit laser beams that can damage your eyes.
2. Align the XFP/10G SFP+ transceiver to the front of the XFP/10G SFP+ transceiver port on the front panel of the appliance.
3. Hold the XFP/10G SFP+ transceiver between your thumb and index finger and insert it into the XFP/10G SFP+ transceiver port, pressing it in until you hear the transceiver snap into place.
4. Move the locking hinge to the DOWN position as shown in the following figure.
   Figure 7. Locking an XFP transceiver



5. Verify that the LED is green and blinks twice, which indicates that the transceiver is functioning correctly.
6. Do not remove the dust caps attached to the transceiver and cable until you are ready to insert the cable.

To remove an XFP/10G SFP+ transceiver

1. Disconnect the cable from the XFP/10G SFP+ transceiver. Replace the dust cap on the cable before putting it away.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Unlock the XFP/10G SFP+ transceiver by moving the locking hinge to the UP position.
3. Hold the XFP/10G SFP+ transceiver between your thumb and index finger and slowly pull it out of the port.
4. Replace the dust cap on the transceiver before putting it away.
5. Put the XFP/10G SFP+ transceiver into its original box or another appropriate container.

> ## Note
>
> This section applies to the MPX/SDX 14020 40G, MPX/SDX 14040 40G, MPX/SDX 14060 40G, MPX/SDX 14080 40G and MPX/SDX 25100 40G, MPX/SDX 25160 40G appliances.

A Quad Small Form-Factor Pluggable (QSFP+) SR4 transceiver is a compact optical transceiver that can operate at speeds of up to 40 gigabits per second. It is hot-swappable on the NetScaler appliances and is available in the maximum link length of 100 meters. Auto-negotiation is enabled by default on the 40G QSFP+ ports into which you insert your 40G QSFP+ SR4 transceiver. As soon as a link between the port and the network is established, the mode is matched on both ends of the cable, and the link speed is auto-negotiated.

> ## Warning
>
> NetScaler appliances do not support 40G QSFP+ SR4 transceivers from vendors other than Citrix Systems. Attempting to install third-party 40G QSFP+ transceivers on your NetScaler appliance voids the warranty.

Insert 40G QSFP+ SR4 transceivers into the 40G QSFP+ ports on the front panel of the appliance. Follow the removal procedure carefully to avoid damaging the 40G QSFP+ SR4 transceiver or the appliance. Also note that frequent installation and removal of transceivers shortens their life span.

> ## Warning
>
> Do not install the transceivers with the cables attached. Doing so can damage the cable, the connector, or the optical interface of the transceiver.

To install a 40G QSFP+ SR4 transceiver

1. Remove the 40G QSFP+ SR4 transceiver carefully from its box.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Align the 40G QSFP+ SR4 transceiver to the front of the 40G QSFP+ transceiver port on the front panel of the appliance.
3. Hold the 40G QSFP+ SR4 transceiver between your thumb and index finger and insert it into the 40G QSFP+ transceiver port, pressing it in until you hear the transceiver snap into place.

4. Lock the transceiver.
5. Verify that the LED is green and blinks twice, which indicates that the transceiver is functioning correctly.

To remove a 40G QSFP+ SR4 transceiver

1. Disconnect the cable from the 40G QSFP+ SR4 transceiver. If you are using a fiber optic cable, replace the dust cap on the cable before putting it away.
   Danger: Do not look directly into fiber optic transceivers or cables. They emit laser beams that can damage your eyes.
2. Unlock the 40G QSFP+ SR4 transceiver.
3. Hold the 40G QSFP+ SR4 transceiver between your thumb and index finger and slowly pull it out of the port.
4. If you are removing a fiber 40G QSFP+ SR4 transceiver, replace the dust cap before putting it away.

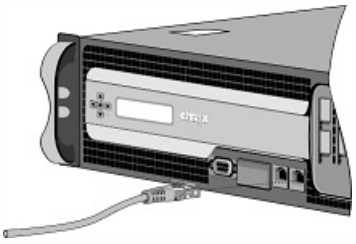
When the appliance is securely mounted on the rack, you are ready to connect the cables. Ethernet cables and the optional console cable are connected first. Connect the power cable last.

Danger: Before installing or repairing the appliance, remove all jewelry and other metal objects that might come in contact with power sources or wires. When you touch both a live power source or wire and ground, any metal objects can heat up rapidly and cause burns, set clothing on fire, or fuse the metal object to an exposed terminal.

Connecting the Ethernet Cables

Ethernet cables connect your appliance to the network. The type of cable you need depends on the type of port used to connect to the network. Use a category 5e or category 6 Ethernet cable with a standard RJ-45 connector on a 10/100/1000BASE-T port or 1G SFP copper transceiver. Use a fiber optic cable with an LC duplex connector with a 1G SFP fiber transceiver, 10G SFP+, or XFP transceiver. The type of connector at the other end of the fiber optic cable depends on the port of the device that you are connecting to.

1. Insert the RJ-45 connector on one end of your Ethernet cable into an appropriate port on the front panel of the appliance, as shown in the following figure.
   Figure 8. Inserting an Ethernet cable



2. Insert the RJ-45 connector on the other end into the target device, such as a router or switch.
3. Verify that the LED glows amber when the connection is established.

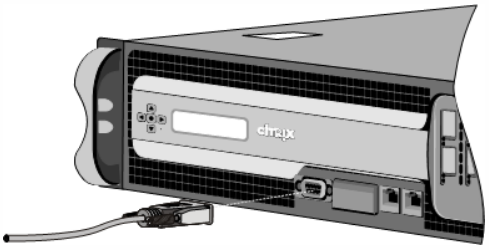
1. Remove the dust caps from the transceiver and cable.
2. Insert the LC connector on one end of the fiber optic cable into the appropriate port on the front panel of the appliance.
3. Insert the connector on the other end into the target device, such as a router or switch.

4. Verify that the LED glows amber when the connection is established.

Connecting the Console Cable

You can use the console cable to connect your appliance to a computer or terminal, from which you can configure the appliance. Alternatively, you can use a computer connected to the network. Before connecting the console cable, configure the computer or terminal to support VT100 terminal emulation, 9600 baud, 8 data bits, 1 stop bit, parity, and flow control set to NONE. Then connect one end of the console cable to the RS232 serial port on the appliance and the other end to the computer or terminal.

1. Insert the DB-9 connector at the end of the cable into the console port that is located on the front panel of the appliance, as shown in the following figure.
   Figure 9. Inserting a console cable



   Note: To use a cable with an RJ-45 converter, insert the optional converter provided into the console port and attach the cable to it.
2. Insert the RJ-45 connector at the other end of the cable into the serial port of the computer or terminal.

Connecting the Power Cable

An MPX 5500, MPX 5550/5650, MPX 7500/9500, MPX 8005/8015/8200/8400/8600/8800 appliance has one power cable. All the other appliances come with two power cables, but they can also operate if only one power cable is connected, except the MPX 22040/22060/22080/22100/22120 and MPX 24100/24150 platforms which come with four power cables and require two power cables for proper operation. A separate ground cable is not required, because the three-prong plug provides grounding.

1. Connect one end of the power cable to the power outlet on the back panel of the appliance, next to the power supply, as shown in the following figure.
   Figure 10. Inserting a power cable

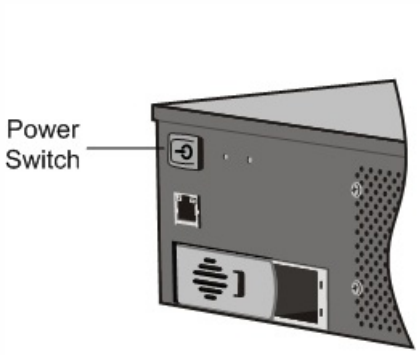
2. Connect the other end of the power cable to a standard 110V/220V power outlet.
3. If a second power supply is provided, repeat steps 1 and 2 to connect the second power supply.
   Note: The MPX 9700/10500/12500/15500, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 17500/19500/21500, and MPX 17550/19550/20550/21550 appliance emit a high-pitched alert if one power supply fails or if you connect only one power cable to the appliance. To silence the alarm,

you can press the small red button located on the back panel of the appliance.

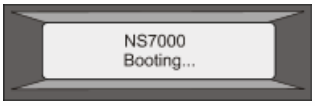
After you have installed the appliance in a rack and connected the cables, verify that the power cable is properly connected. If you have installed a second power supply, make sure the second cable is connected to an outlet for a different circuit than the first. After verifying the connections, you are ready to switch on the appliance.

To switch on the appliance

1. Verify that the appliance is connected through a console or Ethernet port. This will ensure that you can configure the appliance after it is switched on.
2. Press the ON/OFF toggle power switch on the back panel of the appliance.
   Figure 11. Power switch on back panel



3. Verify that the LCD on the front panel is backlit and the start message appears, as shown in the following figure.
   Figure 12. LCD startup screen



Caution: Be aware of the location of the emergency power off (EPO) switch, so that if an electrical accident occurs you can quickly remove power from the appliance.

Initial Configuration

Jan 31, 2011

After you have installed your appliance in a rack, you are ready to perform the initial configuration. Once initial configuration is complete, refer to the specific configuration guides for the features you will be using.

Initial configuration is the same for the multifunction Citrix NetScaler, the dedicated NetScaler Gateway Enterprise Edition, and the dedicated Citrix NetScaler Application Firewall appliances. You can use any of the following interfaces for initial configuration of your appliance:

- First-time use wizard—If you use a web browser to connect to the appliance, you are prompted to enter the network configuration and licensing information, if it is not already specified.
- LCD keypad—You can specify the network settings, but you must use a different interface to upload your licenses.
- Serial console—After connecting to the serial console, you can use the NetScaler command line to specify the network settings and upload your licenses,
- Dynamic Host Configuration Protocol (DHCP)—If you want to configure a new appliance from a remote network, or if you want to install multiple NetScaler appliances and then configure them without using the console port, you can use DHCP to assign each new appliance an IP address at which you can access the appliance for remote configuration.

For initial configuration, use nsroot as both the administrative user name and the password. For subsequent access, use the password assigned during initial configuration.

After you complete the initial configuration of the appliance, you can configure secure access to your appliance. As a result, you are no longer prompted for a password when logging on. This is especially helpful in environments for which you would otherwise have to keep track of a large number of passwords.

This document includes the following details:

To configure a NetScaler appliance (or NetScaler virtual appliance) for the first time, you need an administrative computer configured on the same network as the appliance.

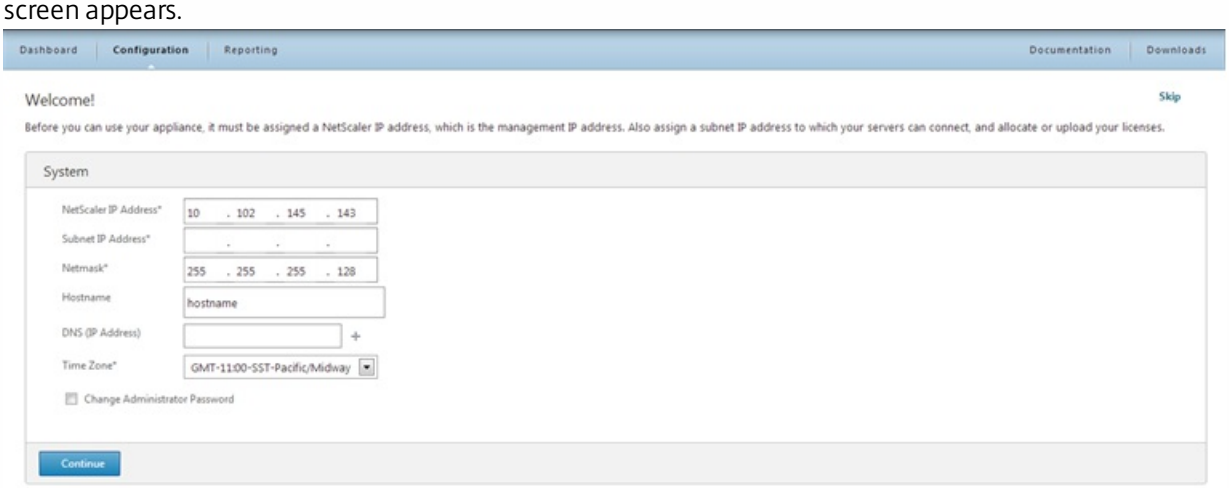
Note: Unless stated otherwise, the term "NetScaler appliance" refers to either a physical or a virtual appliance.
You must assign a NetScaler IP (NSIP) address as the management IP address of your NetScaler appliance, a subnet mask (Netmask), a subnet IP (SNIP) address to which your servers can connect, and a subnet mask (Subnet IP Address Netmask). If you assign a host name, you will be able to access the appliance by specifying its name instead of its NSIP address. Also, assign a time zone.

The wizard automatically appears if any of the following conditions are met:

- The appliance is configured with the default IP address (192.168.100.1).
- A subnet IP address or a mapped IP address is not configured.
- Licenses are not present on the appliance.

To perform first-time configuration of your appliance

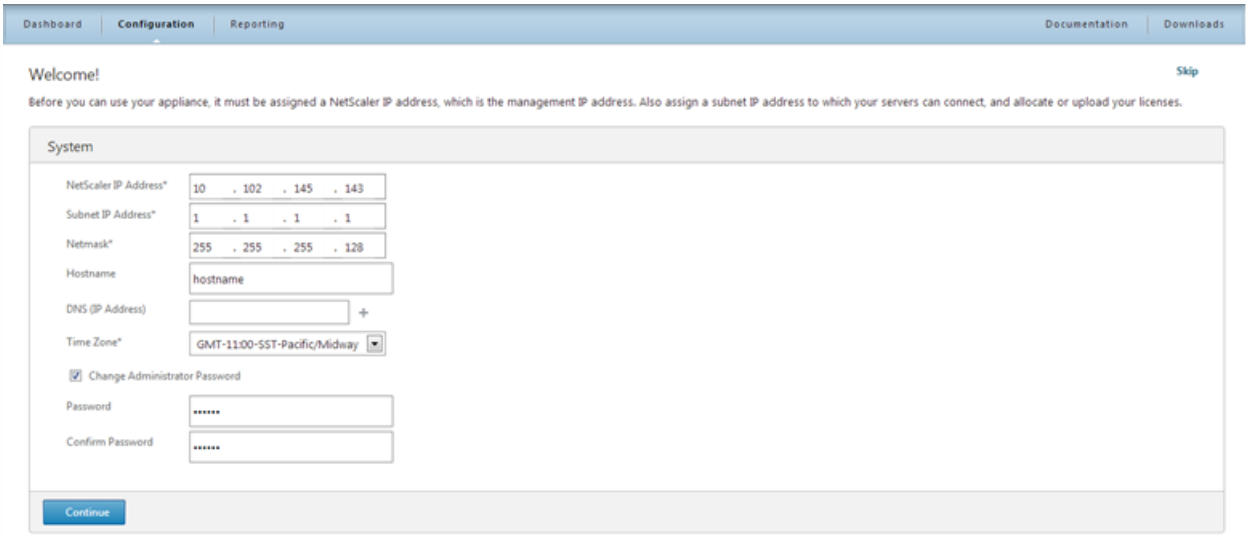
1. In a web browser, type: http://192.168.100.1
   Note: The NetScaler software is preconfigured with a default IP address.
2. In User Name and Password, type the administrator credentials. You can obtain the initial user name and password from your sales representative or from Citrix Customer Service. In Deployment Type, select NetScaler ADC. The following screen appears.



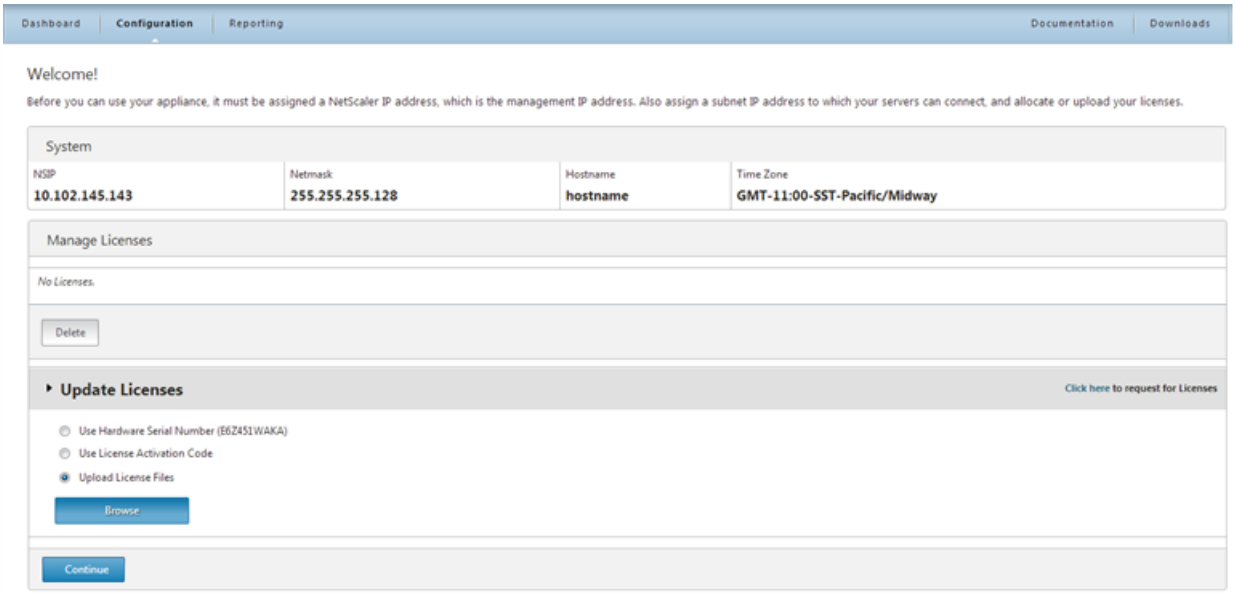
   Enter values for NSIP, SNIP, Netmask, host name, DNSIP and Time Zone. Citrix recommends that you change the administrator password.

   Upload or allocate licenses. Click Add New License and follow the prompts on the screen.

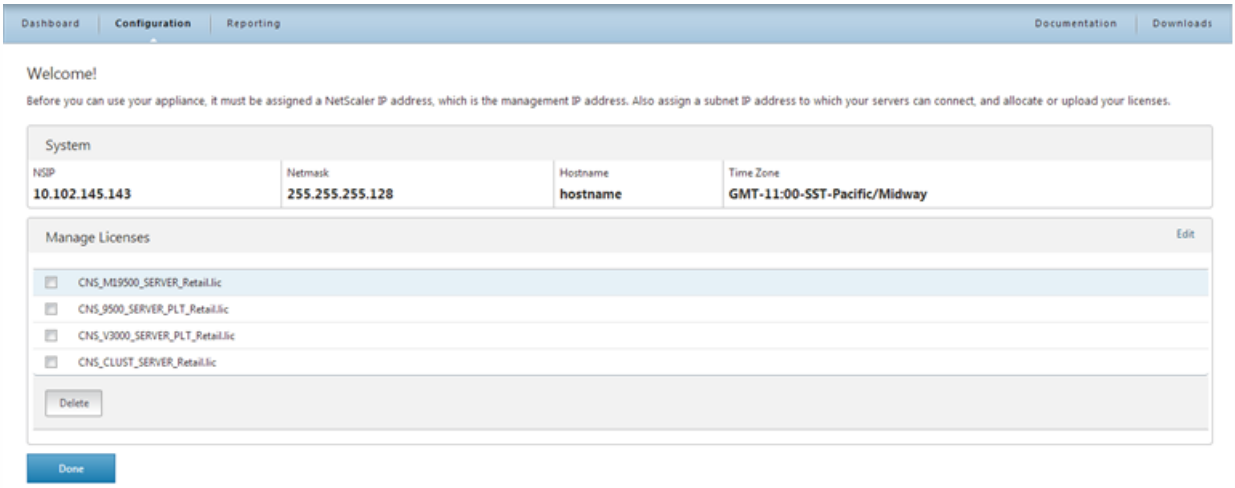
3. Click Continue, as shown in the following screenshot.



4. If you have downloaded your licenses from the licensing portal to your local computer, select Upload License Files, and then click Browse. Navigate to the location of the license files, select the license file, and then click Open.

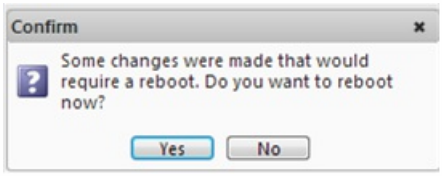


5. If you have not downloaded your licenses to your local computer, click Continue, and then click Done.



The next time you log on to the appliance, you will be prompted to allocate licenses.

6. When prompted, select Reboot.

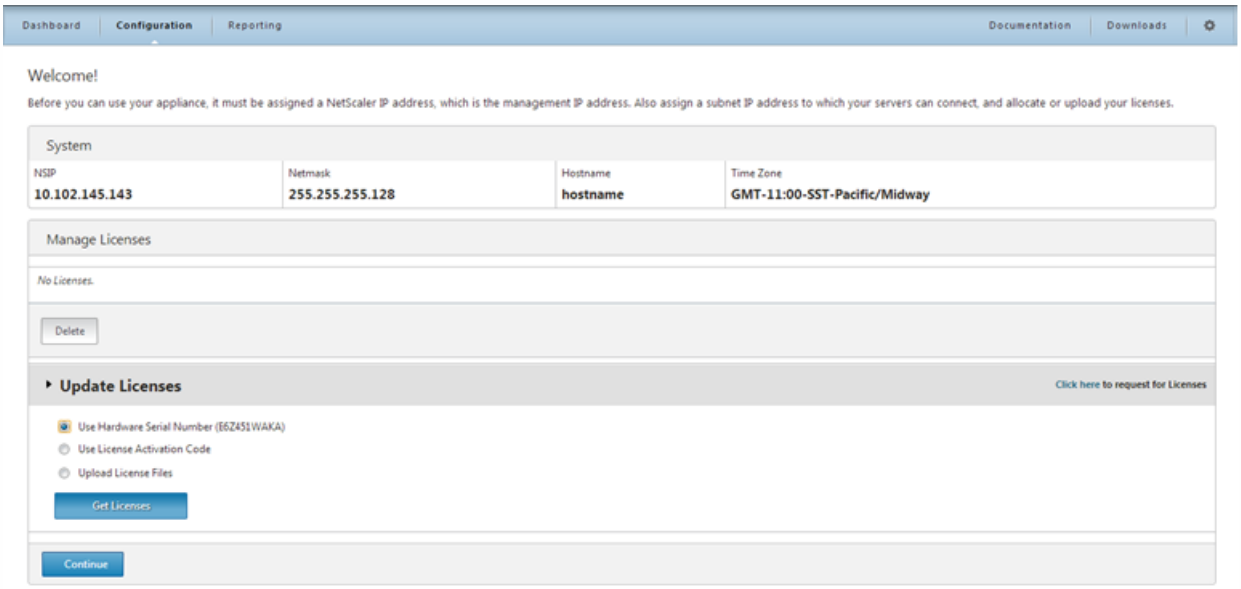


To allocate licenses

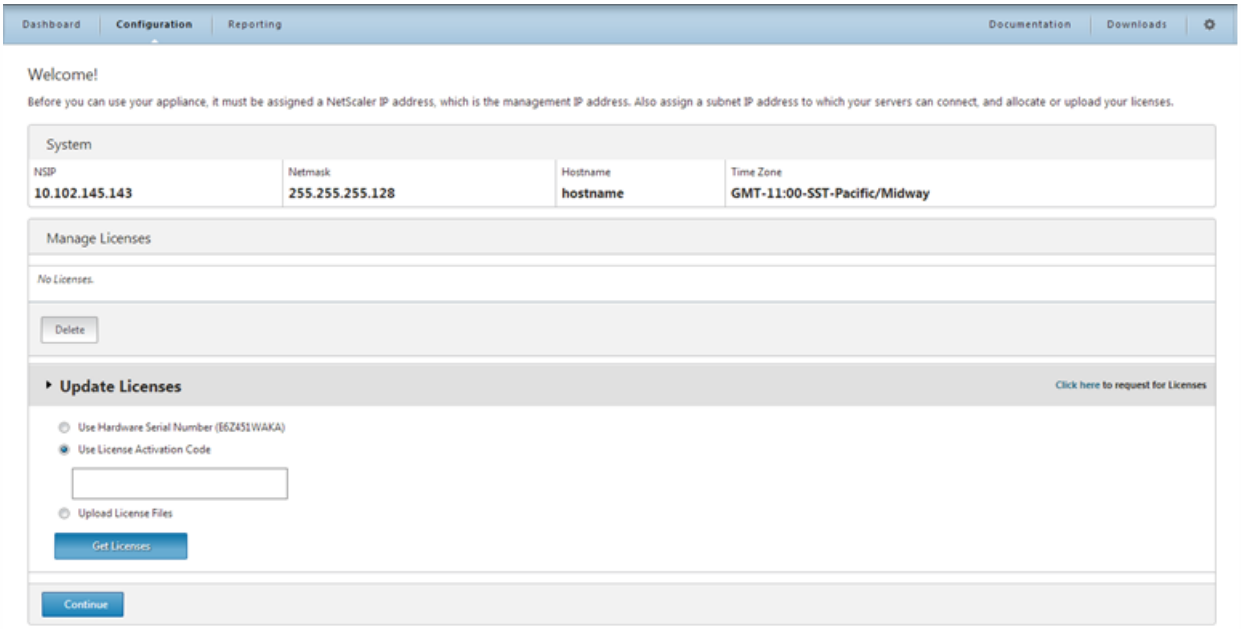
If you did not upload any licenses when you performed first-time configuration of your appliance, you are prompted to allocate licenses the next time you log on to the appliance. Log on and allocate the licenses as follows:

1. In a web browser, type the NSIP address of the appliance.
2. In User Name and Password, type the administrator credentials.
3. The following screen appears (Sample values shown.)

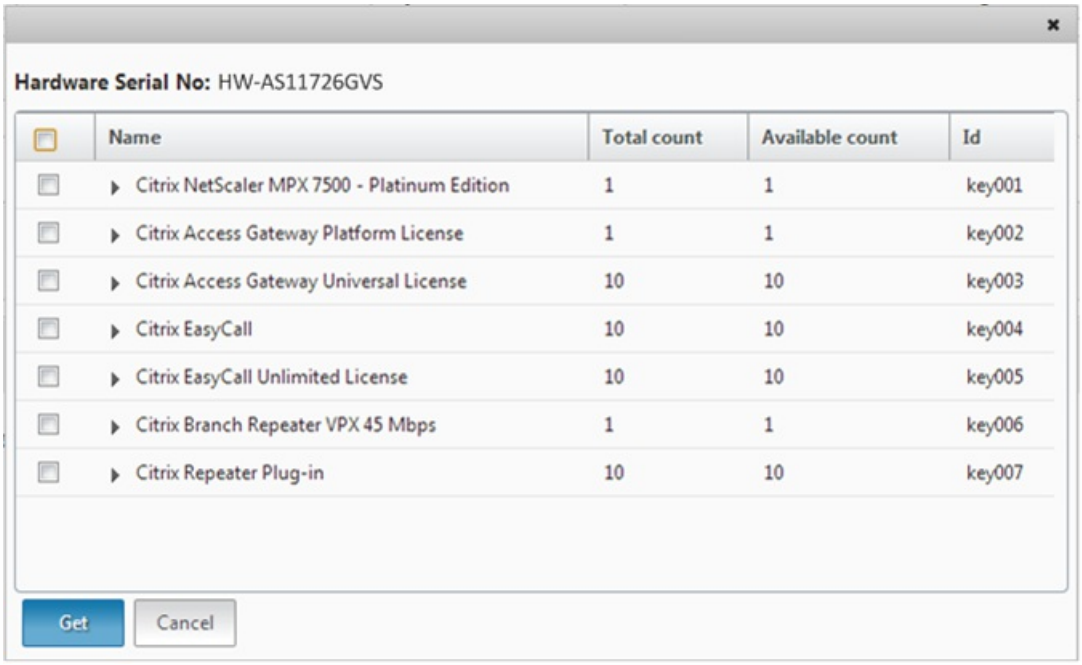
4. Click Continue, and then select one of the following options:
   - **Use Hardware Serial Number**—The software internally fetches the serial number of your appliance and uses this number to display your license(s).



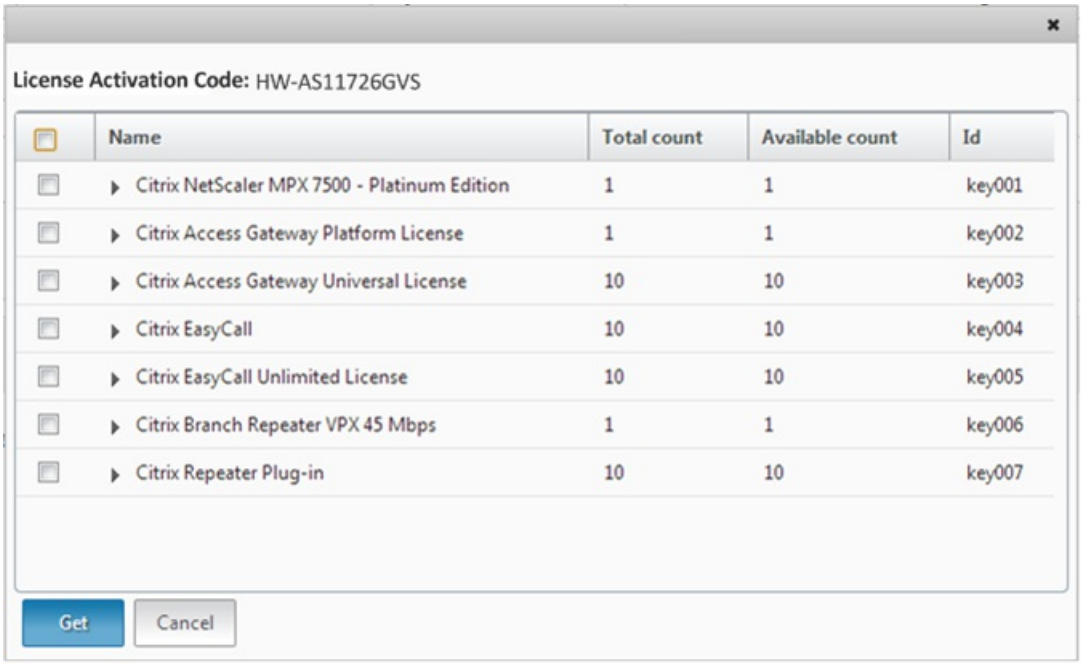
   - **Use License Activation Code**—Citrix emails the LAC for the license that you purchased. Enter the LAC in the text box.



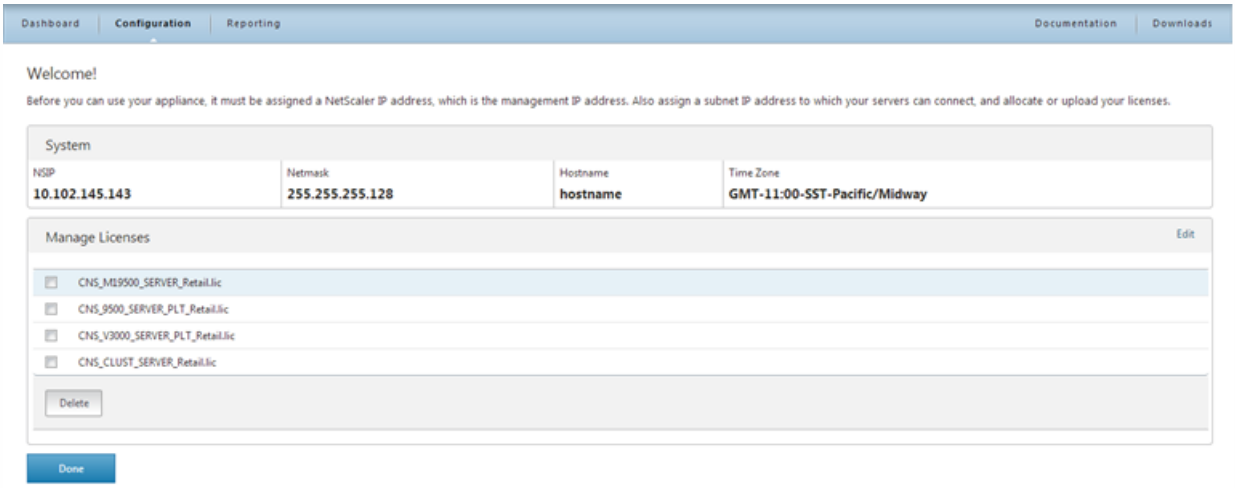
5. Click Get Licenses. Depending on the option that you selected one of the following dialog boxes appears.
   - The following dialog box appears if you selected Hardware Serial Number.



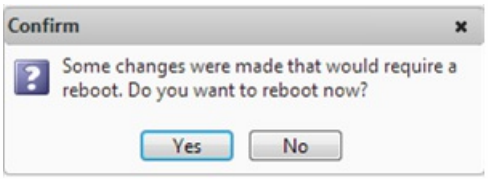
   - The following dialog box appears if you selected Use License Activation Code.



6. Select the license that you want to allocate, and then click Get.
7. Click Continue, and then click Done.



8. When prompted to reboot, select Yes.



When you first install the appliance, you can configure the initial settings by using the LCD keypad on the front panel of the appliance. The keypad interacts with the LCD display module, which is also on the front panel of these appliances.

Note: You can use the LCD keypad for initial configuration on a new appliance with the default configuration. The configuration file (ns.conf) should contain the following command and default values.
set ns config -IPAddress 192.168.100.1 -netmask 255.255.0.0

The functions of the different keys are explained in the following table.

Table 1. LCD Key Functions

Key	Function
<	Moves the cursor one digit to the left.
>	Moves the cursor one digit to the right.
^	Increments the digit under the cursor.
v	Decrements the digit under the cursor.
.	Processes the information, or terminates the configuration, if none of the values are changed. This key is also known as the ENTER key.

To perform the initial configuration by using the LCD keypad press the "<" key.

You are prompted to enter the subnet mask, NetScaler IP address (NSIP), and gateway in that order respectively. The subnet mask is associated with both the NSIP and default gateway IP address. The NSIP is the IPv4 address of the NetScaler appliance. The default gateway is the IPv4 address for the router, which will handle external IP traffic that the NetScaler cannot otherwise route. The NSIP and the default gateway should be on the same subnet.

If you enter a valid value for the subnet mask, such as 255.255.255.224, you are prompted to enter the IP address. Similarly, if you enter a valid value for the IP address, you are prompted to enter the gateway address. If the value you entered is invalid, the following error message appears for three seconds, where xxx.xxx.xxx.xxx is the IP address you entered, followed by a request to re-enter the value.

Invalid addr!
xxx.xxx.xxx.xxx
If you press the ENTER (.) key without changing any of the digits, the software interprets this as a user exit request. The following message will be displayed for three seconds.

Exiting menu...
xxx.xxx.xxx.xxx
If all the values entered are valid, when you press the ENTER key, the following message appears.

Values accepted,
Rebooting...
The subnet mask, NSIP, and gateway values are saved in the configuration file.

Note: For information about deploying a high availability (HA) pair, see "http://support.citrix.com/proddocs/topic/ns-system-10-1-map/ns-nw-ha-cnfgrng-ha-con.html."

When you first install the appliance, you can configure the initial settings by using the serial console. With the serial console, you can change the system IP address, create a subnet or mapped IP address, configure advanced network settings, and change the time zone.

Note: To locate the serial console port on your appliance, see "RS232 Serial Console Port" in "Ports."

To configure initial settings by using a serial console

1. Connect the console cable into your appliance. For more information, see "Connecting the Console Cable" in "Connecting the Cables."
2. Run the vt100 terminal emulation program of your choice on your computer to connect to the appliance and configure the following settings: 9600 baud, 8 data bits, 1 stop bit, parity, and flow control set to NONE.
3. Press ENTER. The terminal screen displays the Logon prompt.
   Note: You might have to press ENTER two or three times, depending on which terminal program you are using.
4. Log on to the appliance with the administrator credentials. Your sales representative or Citrix Customer Service can provide you with the administrator credentials.
5. At the prompt, type config ns to run the NetScaler configuration script.
6. To complete the initial configuration of your appliance, follow the prompts.
   Note: To prevent an attacker from breaching your ability to send packets to the appliance, choose a non-routable IP address on your organization's LAN as your appliance IP address.
   You can replace steps 5 and 6 with the following NetScaler commands. At the NetScaler command prompt, type:

   set ns config -ipaddress<IPAddress> -netmask<subnetMask>

   add ns ip<IPAddress> <subnetMask> -type<type>

   add route<network> <netmask> <gateway>

   set system user <userName> -password

   save ns config

   reboot

   Example

   ```
   set ns config -ipaddress 10.102.29.60 -netmask 255.255.255.0
   add ns ip 10.102.29.61 255.255.255.0 -type snip
   add route 0.0.0.0 0.0.0.0 10.102.29.1
   set system user nsroot -password
   Enter password: *****
   Confirm password: *****
   save ns config
   reboot
   ```

   You have now completed initial configuration of your appliance. To continue configuring the appliance, choose one of the following options:

   Citrix NetScaler.

   If you are configuring your appliance as a standard NetScaler with other licensed features, see "Load Balancing."

   Citrix NetScaler Application Firewall.

   If you are configuring your appliance as a standalone application firewall, see "Application Firewall."

   NetScaler Gateway.

   If you are configuring your appliance as an NetScaler Gateway, see "NetScaler Gateway 10.1."

Note: For information about deploying a high availability (HA) pair, see "Configuring High Availability."

Note: The terms NetScaler, NetScaler appliance, and appliance are used interchangeably.
For initial configuration of a NetScaler appliance, Dynamic Host Configuration Protocol (DHCP) can eliminate dependency on the console by providing a subnet IP (SNIP) address at which you can access the appliance to configure it remotely. You can also use DHCP after initial configuration if, for example, you want to move a NetScaler to a different subnet.

To use DHCP, you must first specify the NetScaler vendor class identifier on a DHCP server. Optionally, you can also specify the pool of IP addresses from which your NetScaler appliance can acquire an IP address. If a pool is not specified, the address is acquired from the general pool.

A new NetScaler appliance does not have a configuration file. When you connect an appliance without a configuration file to the network, its DHCP client automatically polls the DHCP server for an IP address. If you have specified the NetScaler vendor class identifier on the DHCP server, the server returns an address. You can also enable the DHCP client on a previously configured appliance.

Prerequisites

To use DHCP, you must:

1. Note the system ID (sysid) on the serial number sticker on the back panel of the appliance. On an older appliance, the system ID may not be available. In this case, use the MAC address instead of the system ID.
2. Set up a DHCP server and configure it with the NetScaler vendor class identifier.

To configure a Linux/UNIX DHCP server for the NetScaler appliance

1. Specify "citrix-NS" as the vendor class identifier for the NetScaler appliance by adding the following configuration to the server's dhcpd.conf file. The subclass declaration must be inside the subnet declaration.

```
option space auto;
 option auto.key code 1 = text;

class "citrix-1" {
 match option vendor-class-identifier;
}

subclass "citrix-1" "citrix-NS"{
vendor-option-space auto;
option auto.key "citrix-NS";
```

   Note: The location of the dhcpd.conf file can be different in different versions and flavors of the Linux/UNIX-based operating system (for example, in FreeBSD 6.3 the file is present in the /etc/ folder). For the location, see the dhcpd man page of the DHCP server.
2. If you do not want NetScaler appliances to use IP addresses from the general pool, specify a pool of addresses for the appliance. You must include this pool declaration inside the subnet declaration. For example, adding the following configuration to the dhcpd.conf file specifies a pool of IP addresses ranging from 192.168.2.120 to 192.168.2.127.

```
pool {
allow members of "citrix-1";
range 192.168.2.120 192.168.2.127;
option subnet-mask 255.255.255.0;
```


}

3. Terminate the DHCP process and restart it to reflect the change to the configuration file. At the shell prompt, type:
   killall dhcpd

   dhcpd&

```
option space auto;
option auto.key code 1 = text;

class "citrix-1" {
    match option vendor-class-identifier;
}

subnet 192.168.2.0 netmask 255.255.255.0 {
option routers10.217.242.1;
option domain-name"jeffbr.local";
option domain-name-servers8.8.8.8;
default-lease-time 21600;
max-lease-time 43200;
subclass "citrix-1" "citrix-NS" {
vendor-option-space auto;
option auto.key "citrix-NS";
}
pool {
allow members of "citrix-1";
range 192.168.2.120 192.168.2.127;
option subnet-mask 255.255.255.0;
}
}
```

Implementing an Initial NetScaler Configuration from a Remote Computer

When a new NetScaler appliance (or any appliance that does not have a configuration file) starts, it automatically polls the DHCP server for an IP address and provides the DHCP server with its sysid. The DHCP server selects one IP address from its pool and assigns it as a subnet IP (SNIP) address to the appliance. The DHCP server includes the sysid of the appliance and the IP address that it assigns to the appliance in the server's dhcpd.leases file. To find the IP address currently assigned to your appliance, look in the dhcpd.leases file for the last entry with the sysid of your appliance in the uid or client-hostname field. Verify that the binding state in this entry is active. If the binding state is not active but free, the IP address is not yet associated with the appliance.

You can use this address to connect to the appliance and remotely configure the initial settings. For example, you can change the IP address, subnet mask, and gateway settings that were fetched from the DHCP server. After completing the initial configuration, you can manually return the DHCP IP address to the server pool. Alternatively, restarting the appliance automatically releases the DHCP IP address back to the server pool.

You can find out the SNIP address assigned to the appliance from the NetScaler console or from the DHCP server.

At the console prompt, type:

```
> sh dhcpParams
DHCP Client on next reboot is ON
DHCP Client Current State: Active
DHCP Client Default route save: OFF
DHCP acquired IP:192.168.2.127
DHCP acquired Netmask:255.255.255.0
DHCP acquired Gateway:192.168.2.1
Done
```

Look in the dhcpd.leases file for the last entry with the sysid of your appliance in the uid or client-hostname field.

Example: The following entry in a DHCP server's dhcpd.leases file verifies the binding state of the appliance whose sysid is 45eae1a8157e89b9314f.

```
lease 192.168.2.127 {
  starts 3 2013/08/19 00:40:37;
  ends 3 2013/08/19 06:40:37;
  cltt 3 2013/08/19 00:40:37;
  binding state active;
  next binding state free;
  hardware ethernet 00:d0:68:11:f4:d6;
  uid "45eae1a8157e89b9314f";
  client-hostname "45eae1a8157e89b9314f";
```

In the above example, the binding state is ACTIVE and the IP address assigned to the appliance is 192.168.2.127.

The following table describes DHCP-related CLI commands that you might want to use when configuring a new NetScaler appliance.

Table 2. NetScaler CLI commands for using DHCP with a new NetScaler Appliance

Task	At the NetScaler command prompt, type:
To verify the DHCP fetched details, such as IP address, subnet mask, and gateway on the appliance	> sh dhcpParams
To release the DHCP IP address and return it to the IP address pool on the DHCP server when the NetScaler configuration is complete	> release dhcpIP

Using DHCP When a Configuration File is Present

If you need to move a NetScaler appliance to a different subnet, such as from a testing environment to a production environment, you can use DHCP to access an appliance that already has a configuration file. Before moving the appliance,

enable its DHCP client and save the configuration. As a result, when the appliance restarts, it automatically polls the DHCP server for an IP address. If you did not enable the DHCP client and save the configuration before shutting down the appliance, you will need to connect to the appliance through the console and dynamically run the DHCP client on the appliance. The DHCP server will then provide an IP address, a gateway, and a subnet mask. You can use the IP address to access the appliance and configure the other settings remotely.

If the DHCP client is enabled in the configuration file, you should disable it and then save the configuration file. If the DHCP client is enabled, the appliance will poll the DHCP server again for an IP address when it restarts.

The following table lists the NetScaler CLI commands associated with each task.

Table 3. NetScaler CLI commands for using DHCP with a previously configured NetScaler Appliance

Task	At the NetScaler command prompt, type:
To dynamically run the DHCP client to fetch an IP address from the DHCP server	> set dhcpParams dhcpClient on
To configure the DHCP client to run when the appliance restarts	> set dhcpParams dhcpClient on > save config
To prevent the DHCP client from running when the appliance restarts	> set dhcpParams dhcpClient off > save config Note: This is required only if the ON setting was saved.
To save the DHCP acquired route so that it is available when the appliance restarts	> set dhcpParams -dhcpclient on -saveroute on > save config
To prevent saving the DHCP acquired route (default behavior)	> set dhcpParams -dhcpclient on -saveroute off > save config Note: This is required only if the ON setting was saved.

If you administer a large number of NetScaler appliances, storing and looking up passwords for logging on to individual appliances can be cumbersome. To avoid being prompted for passwords, you can set up secure shell access with public key encryption on each appliance.

NetScaler features can also use SSH key based authentication for internal communication when the internal user is

disabled (by using the set ns param -internaluserlogin disabled command). In such cases, the key name must be set as "ns_comm_key".

To set up access using SSH keys, you must generate the public-private key pair on a client and copy the public key to the remote NetScaler appliance.

To generate the keys and connect to a remote NetScaler by using SSH keys

1. On a client (Linux client or a NetScaler) change directory to /root/.ssh.
   cd /root/.ssh
2. Generate the public-private key pair.
   ssh-keygen -t <key_type> -f <optional_key_file_name>

   Example: To create an RSA key with default file name.
   ssh-keygen -t rsa
3. Press ENTER when prompted for a file name for the key pair.
   Note:
   - If you update the default file name for the key pair, use the new name instead of the default name in the rest of this procedure.
   - If you want to disable internal user login, use "ns_comm_key" as the file name for the public-private key pair.
4. Press ENTER two times when prompted for a passphrase.
   Note: If the client is a NetScaler appliance, move the private key file to a persistent location such as sub-directories of the /flash and /var directories.
5. Log on to the remote NetScaler appliance from the client by using a file transfer protocol, and perform the following:
   1. Change directory to /nsconfig/ssh. At the prompt, type:
      cd /nsconfig/ssh
   2. Use the binary transfer mode to copy the public key to this directory.
      bin
      put id_rsa.pub
6. Open a connection to the remote NetScaler appliance by using an SSH client, such as PuTTY, and perform the following:
   1. Log on to the remote appliance using the administrator credentials.
   2. Go to the NetScaler shell.
      > shell
   3. At the shell prompt, change the directory to /nsconfig/ssh.
      root@ns# cd /nsconfig/ssh
   4. Append the public key to the authorized_keys file. At the shell prompt, type:
      root@ns# cat id_rsa.pub >> authorized_keys
      Note: If the authorized_keys file does not exist at the appliance, you need to first create the file and then append the contents.
   5. Change the permission of the /flash, nsconfig, and ssh directories to 755.
      root@ns# chmod 755 /flash
      root@ns# chmod 755 /flash/nsconfig
      root@ns# chmod 755 /flash/nsconfig/ssh
   6. Change the permission of the authorized_keys file to 744.
      root@ns# chmod 744 authorized_keys
   7. Optionally, remove the public key.
      root@ns# rm id_rsa.pub

7. On the client, verify that you can connect to the remote NetScaler appliance by using SSH, without entering the password.
   If using the default file name for the public-private key pair.
   ssh <user_name>@<NetScalerIPAddress>
   If using "ns_comm_key" (when internal user is disabled) for the public-private key pair.
   ssh –i /nsconfig/ssh/ns_comm_key <user_name>@<NetScalerIPAddress>
   If using any other name for the public-private key pair.
   ssh –i <path_to_client_private_key> <user_name>@<NetScalerIPAddress>

The default user account is the administrative account, which provides complete access to all features of the Citrix NetScaler appliance. Therefore, to preserve security, the administrative account should be used only when necessary, and only individuals whose duties require full access should know the password for the administrative account. The default administrative username and password are nsroot and nsroot, respectively. Citrix recommends changing the administrative password frequently.

To change the administrative password by using the configuration utility

1. Log on to the appliance by using the administrative credentials.
2. On the Configuration tab, in the navigation pane, expand System, and then click Users.
3. In the Users pane, click the default user account (nsroot), and then click Change Password.
4. In the Change Password dialog box, in Password and Confirm Password, type the password of your choice.
5. Click OK.

To change the administrative password by using the command line interface

At the command prompt, type:

set system user <userName> -password

Example:

```
set system user nsroot -password
Enter password: *****
Confirm password: *****
Done
```

Lights Out Management Port of the NetScaler MPX Appliance

Dec 17, 2015

The MPX 8005/8015/8200/8400/8600/8800, MPX 11500/13500/14500/16500/18500/20500, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, and MPX 25100T/25160T appliances have an Intelligent Platform Management Interface (IPMI), also known as the Lights Out Management (LOM) port, on the front panel of the appliance. You can use the LOM port to remotely monitor and manage the appliance, independently of the NetScaler software.

By connecting the LOM port to a dedicated channel that is separate from the data channel, you can make sure that connectivity to the appliance is maintained even if the data network is down. You thereby eliminate the data cable and data network as a single point of failure.

You can access the LOM port through a browser and use the graphical user interface (GUI) for most tasks. All tasks can be performed through the NetScaler shell.

You can use either the GUI or a shell for the following tasks:

- Configuring the network settings
- Health monitoring
- Power control operations
- Factory reset

Different Citrix appliances support different shells:

- For FreeBSD based NetScaler MPX appliances, use the bash nsroot shell (also known as NS Shell).
- For Linux based appliances, use the Linux bash root shell.

Note: The terms LOM and Baseboard Management Controller (BMC) are used interchangeably.
Caution: LOM firmware versions are platform specific. Upgrading to a LOM firmware version other than one shown for your platform in the LOM Support Matrix, below, results in the LOM becoming unusable.
The LOM Support Matrix shows the LOM firmware versions shipped with the various platforms, along with the recommended versions, and the earliest NetScaler software versions that support both the shipped and the recommended LOM firmware versions. The latest available LOM package can be found on the Citrix downloads website under LOM Firmware Upgrade.

Hardware	Ships With Version	Recommended Version	Minimum NetScaler Version to avoid PS failure issues
MPX 8005/8015/8200/8400/8600/8800	2.04/2.07/3.02/3.10/3.11	3.11	9.3_65.x, 10.1_123.x, 10.5
MPX 11500/13500/14500/16500/18500/20500	2.52/3.02/3.33/3.39	3.39	9.3_65.x, 10.1_123.x, 10.5
MPX 11515/11520/11530/11540/11542	2.52/3.02/3.33/3.39	3.39	9.3_65.x, 10.1_123.x, 10.5

Hardware	Ships With Version	Recommended Version	Minimum NetScaler Version to avoid PS failure issues
MPX 17550/19550/20550/21550	2.52/3.02/3.33/3.39	3.39	9.3_65.x, 10.1_123.x, 10.5
MPX 22040/22060/22080/22100/22120	2.63/3.22	3.22	9.3_65.x, 10.1_123.x, 10.5
MPX 24100/24150	2.63/3.22	3.22	9.3_65.x, 10.1_123.x, 10.5

Configuring the Network Settings on the LOM Port

Jan 31, 2011

The default IP address for initial access to the LOM port is 192.168.1.3. Change the default credentials and IP address the first time you log on. All LOM GUI operations require you to connect to the appliance by typing the LOM IP address in a web browser and then entering the administrator credentials. Alternatively, you can access LOM functionality through the command line by using the *ipmitool* utility. Using the ipmitool utility remotely, you can determine the LOM firmware version number, perform warm and cold restarts, configure LOM network settings, monitor the health of the appliance, and perform power control operations. The utility is available for download at http://ipmitool.sourceforge.net/. The ipmitool utility is also included in NetScaler MPX and CloudBridge/SDX (dom0) appliances for initial LOM port network configuration. When using the shell, you can choose to use DHCP or static IP settings for initial network configuration. After configuring the network settings, you can use the ipmitool commands over the network. For example, the BMC firmware revision command would need the same username, password, and IP address that is used to access the BMC/LOM GUI port.

For initial configuration, connect the network port on your laptop or workstation directly to the LOM port with a crossover cable, or to a switch in the same local subnet(192.168.1.x) as the LOM port. Assign a network-reachable IP address and change the default credentials. After saving the new settings, the LOM restarts and the changes take effect. After the restart, you must use the new address to access to the LOM.

If you make a mistake that results in losing network connectivity at both the old and new IP addresses, you must use the local shell method to recover.

See the Secure Deployment Guide for best practices for managing administrative credentials and configuring your network for a secure LOM deployment.

Note: On all MPX platforms, except MPX 22040/22060/22080/22100/22120 and MPX 24100/24150, the LEDs on the LOM port are nonoperational by design.
Tip: For first-time setup in a network, to facilitate troubleshooting, make sure that a laptop/PC is connected directly to the LOM port. If you can ping and access the LOM GUI at the default IP address (192.168.1.3) by using static addressing on the laptop/PC, but remote access does not work, take a closer look at network firewall settings and access control list (ACL) policies of all network devices along the network path.
Tip: If some LOM GUI features work but others do not, (for example, normal NetScaler console output is visible in the NetScaler console window in the LOM GUI, but typing in the console does not work), try the above method to isolate the cause to the specific BMC protocol being blocked by the network.
Tip: Some LOM GUI features, such as the NetScaler console, require the latest Java security updates on the laptop/PC. Make sure that the latest Java updates are installed on your laptop/PC.

1. In a web browser, type http://192.168.1.3 and enter the default user credentials.
   Note: The NetScaler LOM port is preconfigured with IP address 192.168.1.3 and subnet mask 255.255.255.0.
2. On the Configuration tab, click Network and type new values for the following parameters:
   - IP Address—IP address of the LOM port
   - Subnet Mask—Subnet mask used to define the subnet of the LOM port
   - Default Gateway—IP address of the router that connects the LOM port to the network
3. Click Save.
4. If you want to change the user credentials, navigate to Configuration > Users, select the user, click Modify User, and change the credentials.

1. Configure the IP addressing mode:
   - To use DHCP, at the shell prompt, type:
     ipmitool lan set 1 ipsrc dhcp

     No further IP-level configuration is required.

   - To use static addressing, at the shell prompt, type:
     1. **ipmitool lan set 1 ipsrc static**
     2. **ipmitool lan set 1 ipaddr <LOM IP address>**
     3. **ipmitool lan set 1 netmask <netmask IP address>**
     4. **ipmitool lan set 1 defgw ipaddr <default gateway IP address>**

     The BMC reboots to apply the changes. Pings to the BMC should succeed after approximately 60 seconds.

2. Optionally, to configure Ethernet VLAN ID and priority, at the NetScaler shell prompt type:
   - **ipmitool lan set 1 vlan id <off|<ID>>**
   - **ipmitool lan set 1 vlan priority <priority>**

   You can either disable or enable the VLAN. Set the VLAN ID to a value from 1 to 4094, and the VLAN priority to a value from 0 to 7. After the network settings have been correctly applied, you can access the ipmitool remotely from a physically separate machine over the network. For remote access, enter the BMC username, BMC password, and the BMC IP address. For example, to run the "ipmitool mc info" command, at the shell prompt on a remote machine, type:

   ipmitool –U <username> –P <password> –H <bmc IP address> mc info

There are two NetScaler MIBs: the NetScaler software management MIB and the NetScaler IPMI LOM hardware management MIB. The software management MIB is primarily used for monitoring the application software and the application software's utilization of hardware resources, such as CPU % and memory %. It provides a high level view of the appliance and is therefore suitable for the application monitoring function carried out by an application group within an organization. The LOM MIB is used for monitoring the hardware health and therefore provides a lower level view of the appliance, more applicable to the network monitoring function carried out by a network monitoring group.

The LOM SNMP traps in the LOM MIB report hardware failures. The NetScaler SNMP traps in the NetScaler MIB report software failures and hardware load issues.

The NetScaler MIB has a very small subset of hardware sensors. It does not cover any BIOS level failures, because the BIOS checks the hardware primarily during boot time , before the NetScaler software starts. If the BIOS detects a failure, it does not load the boot loader. If the boot loader does not load, the operating system does not load, and therefore theNetScaler SNMP software service responsible for sending the traps does not load.

The NetScaler Software Management MIB issues a warning under the following conditions only:

1. If the failure is gradual enough for the main CPU to issue an SNMP alert. An electrical failure close to the CPU, such as a failed electrical capacitor, occurs too quickly for the CPU to issue an alert.
2. If the failure happens after the BIOS, Operating System, and SNMP service have started and normal boot-up has been successful.
3. If the failure happens while the operating system and other system software is in a stable enough state for the SNMP software service to run.

Whenever the NetScaler MIB is unable to report these warnings, because of hardware or software failure, the LOM MIB monitors and reports the warnings. The LOM microcontroller operates independently of the NetScaler software. To monitor the hardware and software of the NetScaler appliance, you must use both the NetScaler MIB and the LOM MIB.

The NetScaler IPMI LOM hardware management MIB SNMP firmware runs on the BMC microcontroller chip. The BMC chip CPU sends a warning in the case of a hardware failure, regardless of whether any of the above conditions occurs. For example, if the BIOS halts the system during boot-up because of a memory DIMM failure, the BMC chip uses the BIOS POST code snooping mechanism to detect the failure, and sends a bad DIMM SNMP alert.

You can log on to the LOM port to view the health information about the appliance. All system sensor information, such as system temperature, CPU temperature, and status of fans and power supplies, appears on the sensor readings page. The Event Log records time stamps of routine events such as a power cycle, in addition to recording hardware-failure events. If SNMP traps are enabled, these events can be sent to your SNMP Network Monitoring software. For more information about how to set up an SNMP alert, see Configuring SNMP Alerts.

To obtain health monitoring information

1. In the Menu bar, click System Health.
2. Under Options, click Sensor Readings.

Download the IPMI SNMP management information base (MIB) for your LOM firmware version, and import it into the SNMP monitoring software.

For a sample configuration, see http://www.net-snmp.org/tutorial/tutorial-5/commands/snmptrap.html. For the exact steps of this procedure specific to your environment, contact your SNMP network monitoring software provider.

You can configure SNMP alerts on the LOM. Optionally, you can configure an alert to send emails.

To configure the alerts, you can use the LOM GUI or the NetScaler Shell.

To configure SNMP alerts on the LOM by using the GUI

1. Download the IPMI View utility from **ftp://ftp.supermicro.com/utility/IPMIView/** and install it on your computer. You will use this utility to test the configuration. For more information, see the section about configuring the alert settings in the IPMI View User Guide at http://supermicro.com.
2. Open the IPMI View utility.
3. In the LOM GUI, navigate to Configuration > Alerts, click Alert No 1, and then click Modify.
4. Select the severity level of the events for which to generate alerts.
5. Set Destination IP to the IP address at which you installed the IPMI View utility.
6. Optionally, to receive alerts by email, specify an email address. To avoid receiving email for routine alerts, specify a severity higher than Informational.
7. Click Save.
8. The LOM should start sending alerts to the IPMI View utility within in a minute or two. After the IPMI View utility starts receiving alerts from the LOM, reconfigure the destination IP address to point to your SNMP Network Management Software, such as HP OpenView.

Setting up SNMP Alerts on the LOM by Using the NetScaler Shell

To customize your filter and policy settings, see the IPMI Specification 2.0 rev. 1.1 documentation.

The latest IPMI specifications are available from the IPMI section of the Intel website:

http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-specifications.html

Usually, customization in the SNMP Network Management Software is the preferred method, because it can be done one time at a central location. Therefore, the settings below send all events for all sensors to the SNMP network management software. These are very low traffic events and therefore should not result in any significant network usage.

To set up SNMP filters

The following commands set up SNMP to allow all events:

ipmitool raw 4 0x12 0x6 0x10 0x80 1 1 0 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0 0xff 0 0 0xff 0 0 0xff 0

To set up a policy list

The following command creates a policy list for all sensors and events:

ipmitool raw 4 0x12 9 0x10 0x18 0x11 0x81

To setting up the destination address for SNMP events

The following command sets up a destination IP address for an SNMP event:

ipmitool lan alert set 1 1 ipaddr <x.x.x.x>

Where, <x.x.x.x> is the IP address to which the SNMP event should be sent.

To specify an SNMP community string name

At the prompt, type:

ipmitool lan set 1 snmp <community string>

Installing a Certificate and Key on the LOM GUI

Jan 31, 2011

Citrix recommends using HTTPS to access the LOM GUI. To use HTTPS, you must replace the default SSL certificate with one from a trusted certificate authority and upload a private key to the LOM GUI.

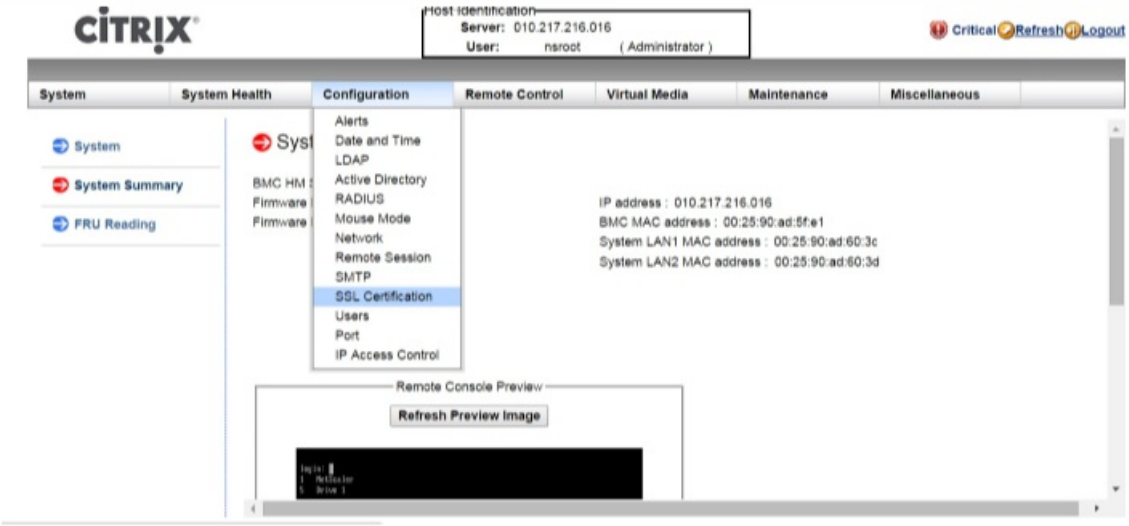
To encrypt SNMP alerts, setup an SSL certificate and private key. In the GUI, navigate to **Configuration** > **SSL Certification** and apply the SSL certificate and private key. See the NetScaler Secure Deployment Guide for more information about how to securely deploy the LOM in your network. To enable encryption and learn the security measures for LOM, see http://support.citrix.com/article/CTX129514.

If you make a mistake, you must restore the BMC to the factory defaults to erase the certificate and key. Use the following shell command:

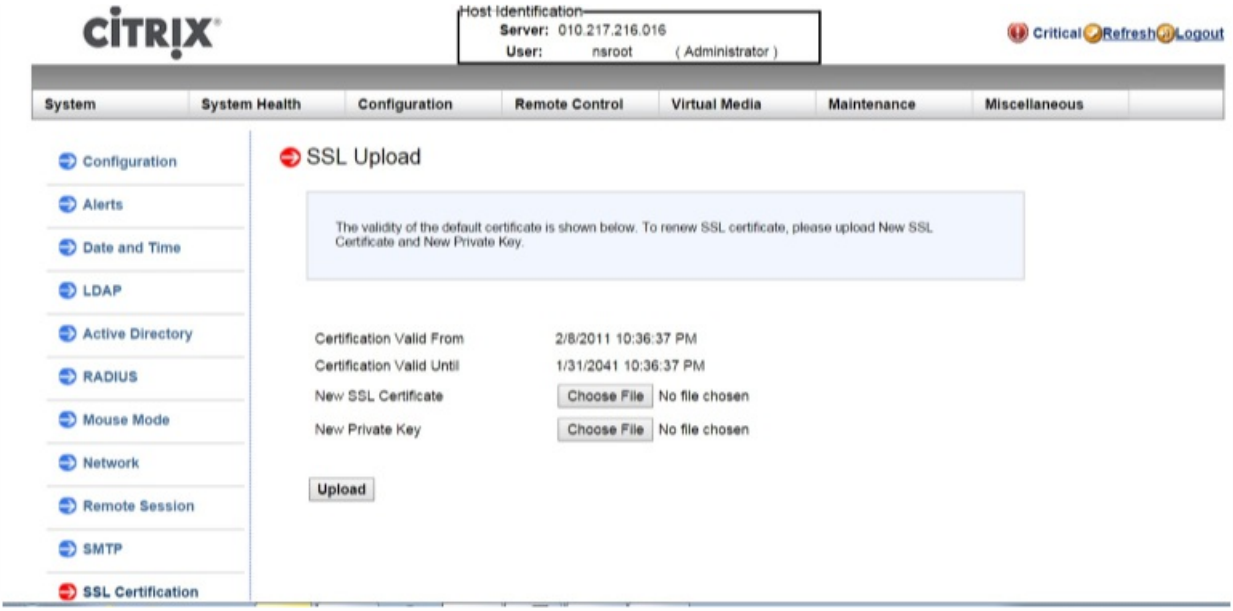
ipmitool raw 0x30 0x41 0x1

Note: The certificate file must contain only the certificate. The certificate and key must not be in the same file. Make sure that the certificate contains only the certificate and that the key file contains only the key.

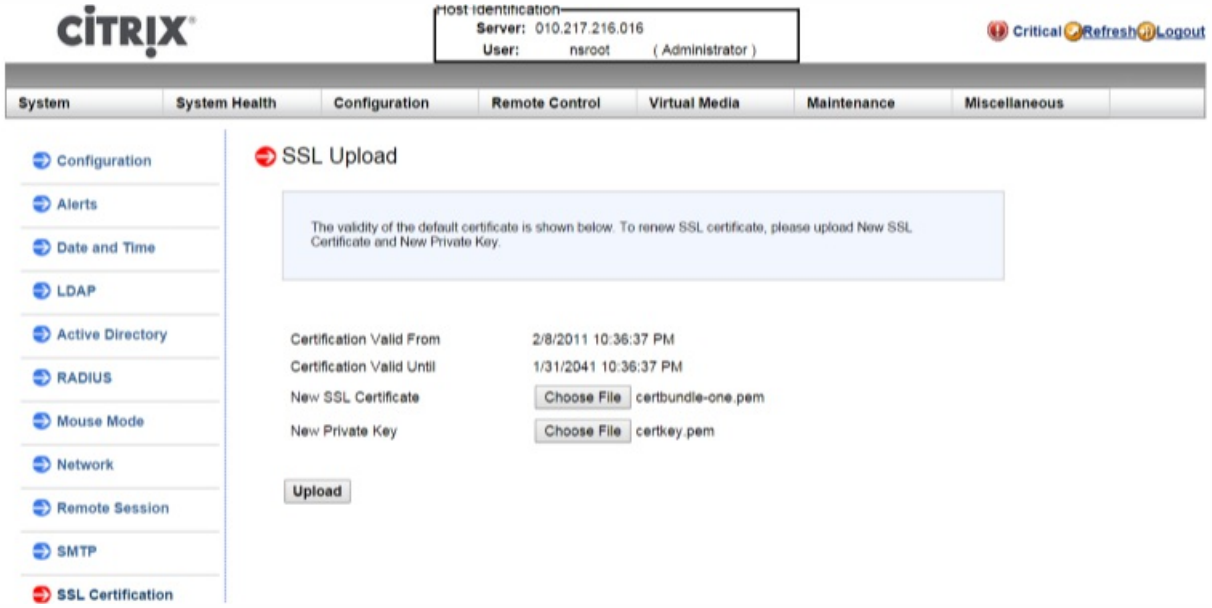
1. Navigate to Configuration > SSL Certification.



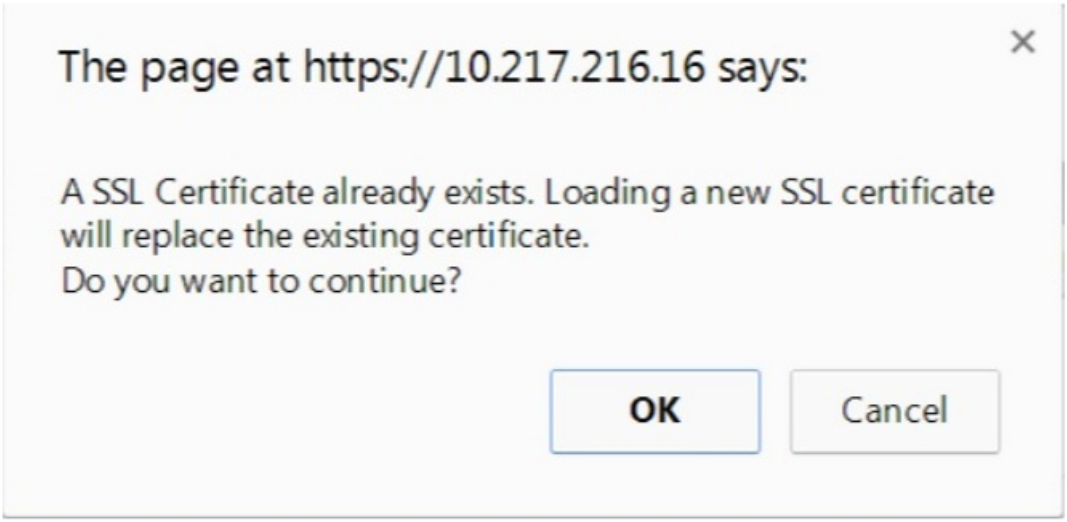
2. In the right pane, click the Choose File buttons to select a new SSL certificate and a new private key.



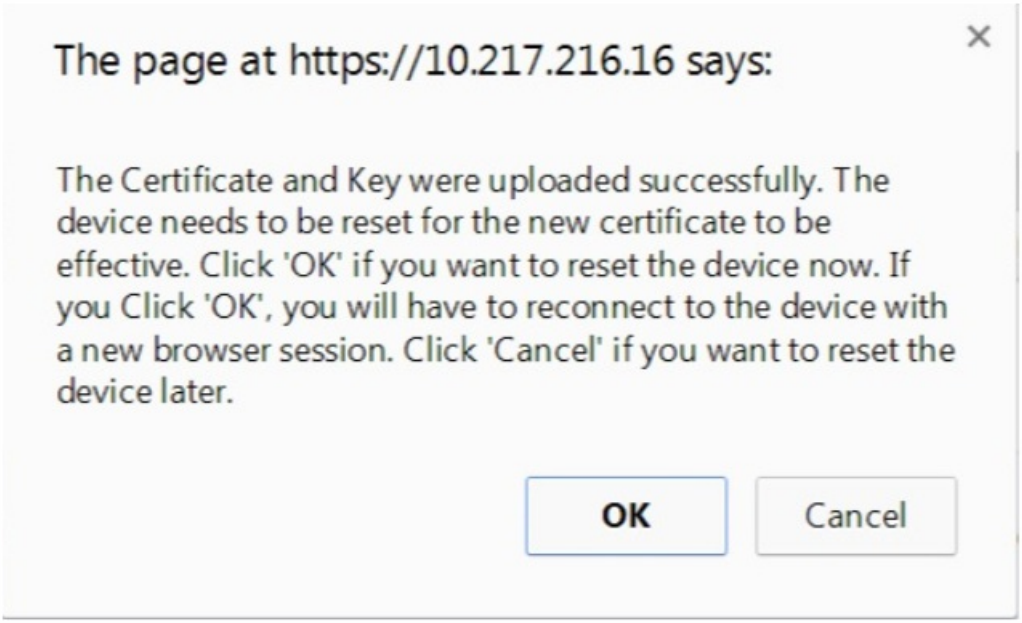
3. To verify that you have selected the correct certificate and private key, check the file names of the certificate and key, which appear next to the Choose File buttons.



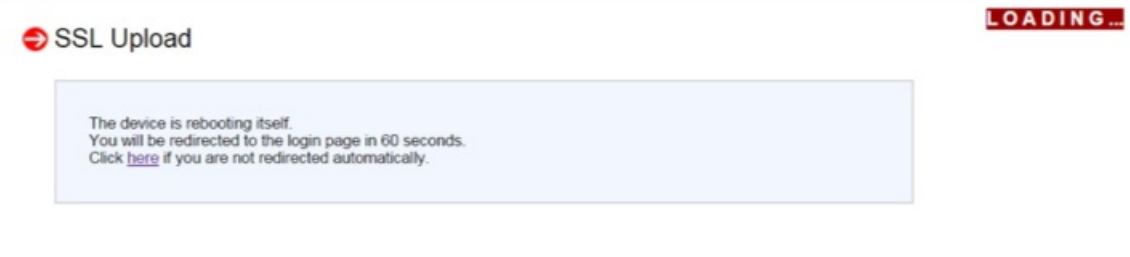
4. Click Upload. A message informs you that uploading a new SSL certificate replaces the existing (default) certificate.
5. Click OK.



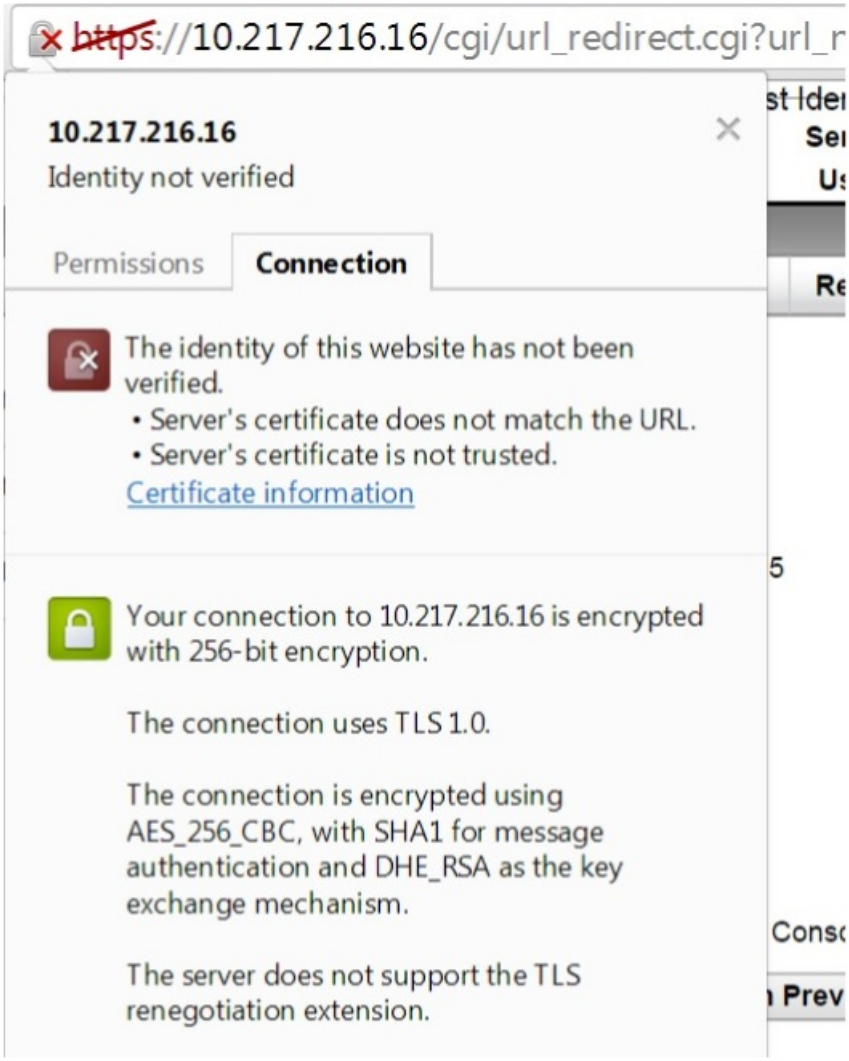
6. When a message informs you that the certificate and key have been uploaded successfully, click OK to reset the device.



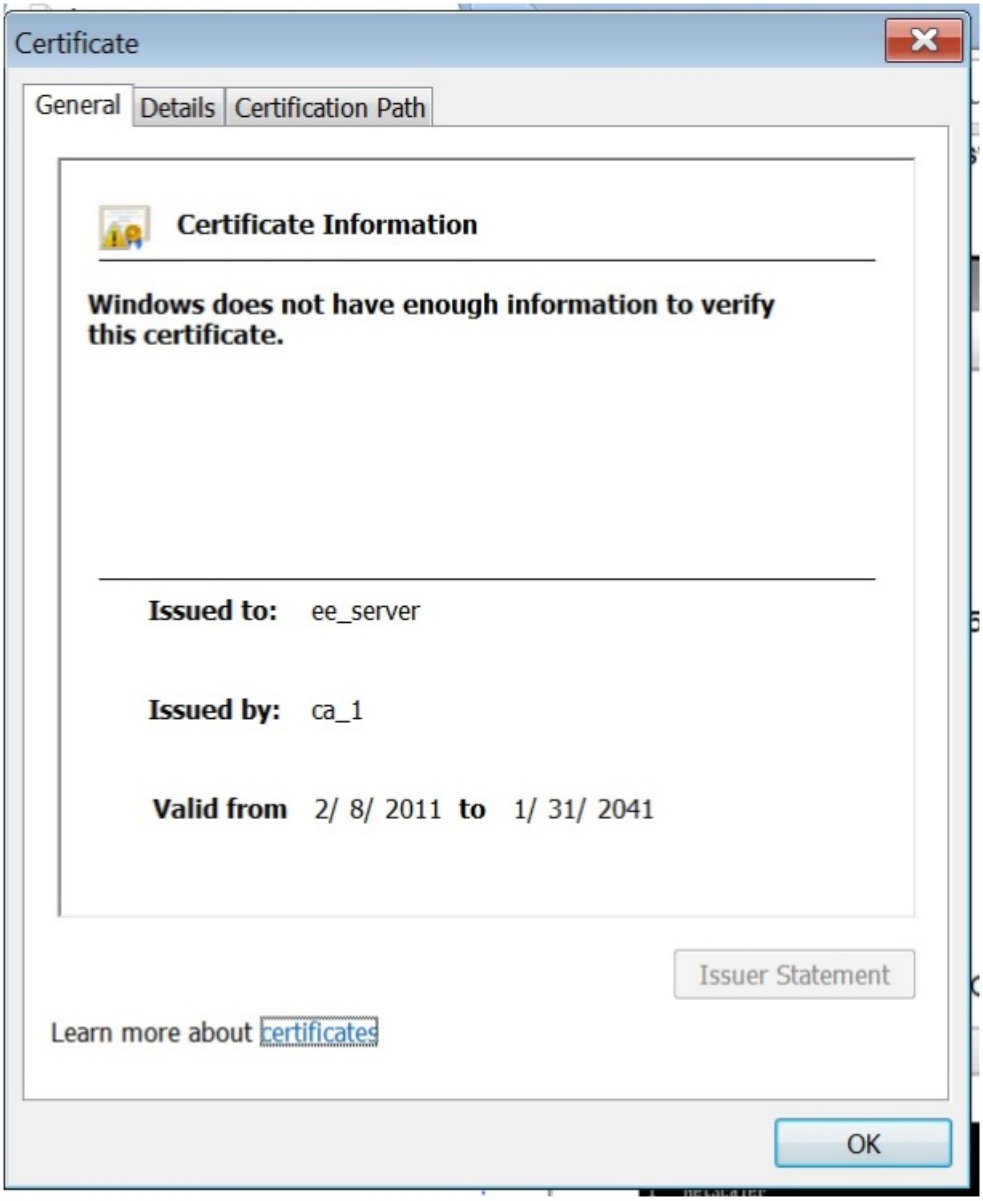
The reset takes approximately 60 seconds. You are then redirected to the logon page.



7. Log on to the LOM GUI by using your default credentials.
   Note: If the certificate or key are invalid, the BMC reboots, tries the new settings, and reverts to using the previous settings.
8. In the address bar, click the lock icon to display the connection tab, as shown on the screen below.



9. Click Certificate information to display details about the certificate that you just uploaded.



Note: For the best practices for LOM and NetScaler security, see http://support.citrix.com/article/CTX129514.

Obtaining the MAC Address, Serial Number, and Host Properties of the Appliance

Jan 31, 2011

A Media Access Control address (MAC address) is a unique identifier assigned to network interfaces for communication on the physical network segment. The serial number is on the back panel of the appliance. If you do not have easy access to the back panel, you can get the appliance's serial number by logging on to the LOM port. You can also retrieve the parameter settings assigned to the IP addresses configured on the appliance, such as the state of ARP, ICMP, telnet, secure shell access, and dynamic routing.

1. In the Menu bar, click Remote Control.
2. Under Options, click Console Redirection.
3. Click Launch Console, and then click Yes.
4. Type the administrator credentials.
5. Type show interface <management_interface_id> to display the MAC address.
6. Type show hardware to display the serial number of the appliance.
7. Type sh nsip to display the host properties of the appliance.

At the shell prompt, type:

ipmitool lan print

Example

```
Set in Progress         : Set Complete
Auth Type Support       : MD2 MD5 OEM
Auth Type Enable        : Callback : MD2 MD5 OEM
                        : User     : MD2 MD5 OEM
                        : Operator : MD2 MD5 OEM
                        : Admin    : MD2 MD5 OEM
                        : OEM      :
IP Address Source       : Static Address
IP Address              : 192.168.1.3
Subnet Mask             : 255.255.255.0
MAC Address             : 00:25:90:3f:5e:d0
SNMP Community String   : public
IP Header               : TTL=0x00 Flags=0x00 Precedence=0x00 TOS=0x00
BMC ARP Control         : ARP Responses Enabled, Gratuitous ARP Disabled
Gratituous ARP Intrvl   : 0.0 seconds
Default Gateway IP      : 0.0.0.0
Default Gateway MAC     : 00:00:00:00:00:00
Backup Gateway IP       : 0.0.0.0
```

```
Backup Gateway MAC     : 00:00:00:00:00:00
802.1q VLAN ID         : Disabled
802.1q VLAN Priority   : 0
RMCP+ Cipher Suites    : 1,2,3,6,7,8,11,12,0
Cipher Suite Priv Max  : aaaaXXaaaXXaaXX
                       :    X=Cipher Suite Unused
                       :    c=CALLBACK
                       :    u=USER
                       :    o=OPERATOR
                       :    a=ADMIN
                       :    O=OEM
```

Performing Power Control Operations by using the LOM Port

Jan 31, 2011

Through the LOM port, you can remotely perform power control operations, such as graceful shutdown and restart, power cycling the appliance, and restarting the BMC microcontroller. A cold restart takes longer than a warm restart. In a cold restart, you switch off power to the appliance and then switch it back on.

1. In the Menu bar, click Remote Control.
2. Under Options, click Power Control, and then select one of the following options:
   - **Reset System**—Gracefully restart the appliance. All operations on the appliance are stopped, no new connections to the client or server are accepted, and all existing connections are closed before the appliance restarts. This is similar to a warm restart, such as by entering the reboot command. The BMC does not reboot itself during this operation.
   - **Power Off System – Immediate**—Disconnect power to the appliance immediately, without gracefully shutting down the appliance. The BMC continues to operate normally in this mode to allow the user to remotely power on the appliance. This is the same as pushing the power button until the unit powers off.
   - **Power Off System – Orderly Shutdown**—Gracefully shut down the appliance, and then disconnect power to the appliance. Has the same effect as pressing the power button on the back panel of the appliance for less than four seconds. All operations on the appliance are stopped, no new connections to the client or server are accepted, and all existing connections are closed before the appliance shuts down. The BMC continues to operate normally in this mode to allow the user to remotely power on the appliance. This is the same as entering the shutdown command in the appliance shell.
   - **Power On System**—Turn on the appliance. The BMC does not reboot itself during this operation. This is the same as pushing the power button.
   - **Power Cycle System**—Turn off the appliance, and then turn it back on. The BMC does not reboot itself during this operation. This is the same as pushing the power button until the unit powers off, and then pushing the power button to power on the unit.
3. Click Perform Action.

A warm restart, cold restart, or a power cycle of the appliance, using the power button, does not include power cycling the BMC. The BMC runs on standby power directly from the power supply. Therefore, the BMC is not affected by any state of the power button on the appliance. The only way to power cycle the BMC is to remove all power cords from the appliance for 60 seconds.

When performing either a warm or cold restart of the BMC microcontroller, you cannot communicate with the LOM port. Both actions restart the BMC but not the main CPU. To perform a warm restart of LOM from the appliance, type:

ipmitool mc reset warm

To perform a warm restart remotely from another computer on the network, type:

ipmitool –U <bmc_gui_username> –P <bmc_gui_password> –H <bmc IP address> mc reset warm

To perform a cold restart of the LOM from the appliance, type:

ipmitool mc reset cold

To perform a warm restart remotely from another computer on the network, type:

ipmitool –U <bmc_gui_username> –P <bmc_gui_password> –H <bmc IP address> mc reset cold

If the appliance fails or becomes unresponsive, you can remotely perform a core dump. This procedure has the same effect as pressing the NMI button on the back panel of the appliance.

To perform a core dump by using the GUI

1. In the Menu bar, click Remote Control.
2. Under Options, click NMI, and then click Initiate NMI.

To perform a core dump remotely from another computer on the network by using the shell

At the shell prompt, type:

ipmitool -U <bmc_gui_username> -P <bmc_gui_password> -H <bmc IP address> chassis power diag

Restoring the BMC Configuration to Factory Defaults

Jan 31, 2011

You can restore the BMC to its factory-default settings, including deleting the SSL Certificate and SSL key.

1. Navigate to Maintenance > Factory Default.
2. Click Restore.

At the shell prompt, type:

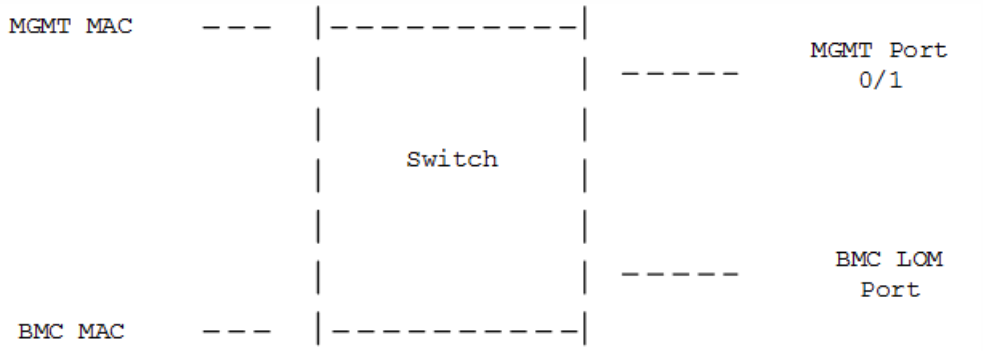
ipmitool raw 0x30 0x41 0x1

Specifying the Port for IPMI BMC Failover

Jan 31, 2011

With LOM firmware version 3.x or later, the default mode for failover between the dedicated LOM port and the shared LOM/management port is to fail over to the active port. By default, no user configuration is needed other than selecting the port to which to connect the cable. The motherboard has an Ethernet switch between the management MAC and the management port, and between the LOM MAC and the LOM port. The following figure shows the Ethernet switch.

Figure 1. Ethernet Switch



You can set this switch to direct LOM traffic through the dedicated LOM port or through the shared management port. A dedicated LOM port removes the management port as a single point of failure, while a shared LOM/management port reduces the cabling costs.

Using the BIOS POST Code to Detect Errors

Jan 31, 2011

You can read the BIOS POST code by using the LOM GUI or the shell. To interpret the BIOS Beep codes, see https://www.ami.com/support/doc/AMI_Aptio_4.x_Status_Codes_PUB.pdf.

Navigate to Miscellaneous > BIOS Post Snooping.

At the prompt, type:

ipmitool raw 0x30 0x2a

Migrating the Configuration of an Existing NetScaler Appliance to Another NetScaler Appliance

Jan 08, 2014

If you are migrating to a new appliance, you must make some changes to the configuration (ns.conf file) of the old appliance before you copy the configuration to the new appliance.

Note: The following procedure does not apply to NetScaler FIPS appliances.

1. On the old appliance, create a backup copy of the configuration file (ns.conf).
2. Use a vi editor to edit the configuration file that you backed up. For example, you might want to change the user name, host name, and password.
   Note: You must remove all interface-related configuration, such as set interface, bind vlan, add channel, bind channel, and set channel.
3. Shut down the old appliance.
4. Perform initial configuration on the new appliance. Connect to the serial console, and at the command prompt type **config ns** to run the NetScaler configuration script. Enter parameter values, such as NetScaler IP address and subnet mask. For information about performing initial configuration by using the configuration utility (GUI) or the LCD keypad, see Initial Configuration.
5. Restart the new appliance.
6. Add a route on the new appliance. At the command prompt, type: add route <network> <netmask> <gateway>
7. Copy the edited configuration file to the new appliance.
8. Copy other relevant files, such as bookmarks, SSL certificates, and CRLs, to the new appliance. Return your feature license(s) to the Citrix licensing portal and reallocate it on the new appliance. For more info about returning your licenses, see http://support.citrix.com/article/CTX131110.
   Note: The platform license is different for a new appliance.
9. Restart the new appliance.
10. Add interface-related configuration specific to your new appliance, switch, and router, and save the configuration.

If you have a high-availability setup, you must perform the above procedure on both the nodes.

Troubleshooting

Feb 28, 2014

NetScaler MPX 8005/8015/8200/8400/8600/8800, MPX 11500/13500/14500/16500/18500/20500, MPX 11515/11520/11530/11540/11542, MPX 17550/19550/20550/21550, MPX 22040/22060/22080/22100/22120, MPX 24100/24150, and MPX 25100T/25160T appliances support LOM. Depending on the state of the LOM configuration, start with one of the steps in the following procedure. (To configure the LOM port, see Lights Out Management Port of the NetScaler Appliance).

1. If the LOM port is configured and known to have been working previously, use the LOM credentials to log on to the LOM GUI, and then do the following:
   1. Navigate to **Remote Control** > **Console Redirection**, and then click **Launch Console**.
   2. On the Java iKVM Viewer screen, check the VGA console window for boot errors, such as bad or missing boot media (boot drive/Compact Flash card), and reseat any unconnected boot media. If the appliance boots up, try to log on and run the show techsupport command from the NetScaler command line. Complete the Check Network Interfaces steps listed below to find a working interface on which to transfer the support bundle file.
   3. Navigate to System Health > Sensor Readings to check the status of the hardware components (for example, CPU temperature, system temperature, and power supply status). You might need to scroll down. Green indicates that the hardware component is functioning properly. Red indicates that it has failed. Contact Citrix Support if you observe red indicators.
   4. Navigate to Miscellaneous > Post Snooping and check for BIOS POST initialization codes. If the value of Post Snooping is "00" or "AC," and the AC power supply LED light is green, the BIOS booted up normally. If not, check the Java iKVM Viewer screen to see if the appliance stopped responding during BIOS POST initialization. Perform substeps a through f of Step 2 to recover the appliance. If these steps fail, contact Citrix Support.
2. If the LOM port is configured and the LOM GUI is not accessible, try pinging the LOM IP address. The baseboard management controller (BMC, also known as LOM) runs on standby power, so even if the appliance is powered off by pressing the power button, the BMC is still working. If you are unable to ping the LOM IP address, connect to the COM1 console port through a serial cable (the serial cable can be connected to a network serial terminal/console server for remote access), or try pinging the NetScaler IP address. On the appliance, do the following:
   1. Verify that the appliance is receiving power.
   2. If the appliance is not receiving power, change the power cable and connect the cable to another socket.
   3. Verify that the power supply is properly seated in power supply slot.
   4. Remove all AC power supply cords for 30 seconds to completely remove power from the appliance.
   5. Reinsert the AC power supply cords and check the LEDs indicating the status of the AC power supplies. If a power-supply LED is not green, troubleshoot the power supply.
   6. Try pinging the LOM IP again. If successful, go to Step 1.
3. If the appliance does not support the LOM port or the LOM port is not configured, do the following:
   1. Connect the serial console cable to the appliance.
   2. Perform the substeps a through e of Step 2.
   3. On the serial console port window, check for any boot failure errors, such as bad or missing boot media (boot drive/Compact Flash card), and reseat any unconnected boot media. If the appliance boots up, try to log on and run the show techsupport command from the NetScaler command line. Complete the Check Network Interfaces steps listed below to find a working interface on which to transfer the support bundle file.

Check Network Interfaces.

1. If management interface 0/1 is not operational, use the Java iKVM Viewer, as described in Step 1.b, to set up management interface 0/2, and connect a network cable to port 0/2. Use the serial console port for appliances that do not support the LOM port.
2. Make sure that the LED port status indicators are green for all interfaces. For more information about LED port status indicators, see "LED Port-Status Indicators" in Ports.
3. Verify that the SFP/SFP+/XFP transceivers are supported by Citrix.

Hardware FAQs

Dec 04, 2014

Are transceivers shipped with the MPX 8005/8015/8200/8400/8600/8800 appliance?

No. Transceivers are available for purchase separately. Contact your Citrix sales representative to order transceivers for your appliance.

Are transceivers hot-swappable?

The 1G SFP transceiver is hot-swappable with release 9.3 build 47.5 or later on the following NetScaler appliances, which use the Intel e1k interface:

- MPX 7500/9500
- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500
- MPX 11500/13500/14500/16500/18500/20500
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150

The 10G SFP+ transceiver is hot-swappable with release 9.3 build 57.5 or later on the following NetScaler appliances, which use the ixgbe (ix) interface:

- MPX 8005/8015/8200/8400/8600/8800
- MPX 9700/10500/12500/15500
- MPX 11500/13500/14500/16500/18500/20500
- MPX 17500/19500/21500
- MPX 17550/19550/20550/21550
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150
- MPX 25100T/25160T

Why does the 10G SFP+ transceiver autonegotiate to 1G speed?

Autonegotiation is enabled by default on the 10G SFP+ ports into which you insert your 10G SFP+ transceiver. When a link is established between the port and the network, the speed is autonegotiated. For example, if you connect the port to a 1G network, the speed is autonegotiated to 1G.

Can I insert a 1G transceiver into a 10G slot?

The 10G slot supports copper 1G transceivers, which can operate at up to 1 Gbps in a 10 Gbps slot.

Note that you cannot insert a 10G transceiver into a 1G slot.

The following table shows the compatibility matrix of transceivers and ports available on the NetScaler appliance.

Ports	Transceivers		
	10G	1G Fiber	1G Copper

10G	Supported	Not Supported	Supported
1G Fiber	Not Supported	Supported	Not Supported
1G Copper	Not Supported	Not Supported	Supported

What is QSFP+?

QSFP+ stands for Quad Small Form-factor Pluggable, which is a small, hot-pluggable transceiver for connecting data devices. This transceiver is used for 40G interfaces.

QSFP+ to Four SFP+ Copper Breakout Cables—These cables connect to four SFP+ 10GE ports of a NetScaler appliance on one end and to a QSFP+ 40G port of a Cisco switch on the other end.

Support for 40G connectivity—NetScaler models that have at least four 10G SFP+ ports connect to Cisco 40G interfaces by aggregating four of the 10G SFP+ ports to form a 40G link aggregation channel. QSFP to Four port SFP+ Copper Breakout Cable **QSFP-4SFP10G-CU3M (reports as L45593-D178-C30)** is used.

Which NetScaler appliances support the QSFP-4SFP10G-CU3M (reports as L45593-D178-C30) Breakout Cable?

NetScaler appliances that have at least four 10G SFP+ ports support this cable. The following appliances have at least four 10G SFP+ ports:

- MPX 11500/13500/14500/16500/18500/20500
- MPX 17550/19550/20550/21550
- MPX 11515/11520/11530/11540/11542
- MPX 22040/22060/22080/22100/22120
- MPX 24100/24150
- MPX 25100T/25160T

QSFP-4SFP10G-CU3M breakout cable is supported by NetScaler release 9.3 build 65.8 or later, and release 10.1 build 122.17 or later.

Is the power supply on the NetScaler MPX 5500 and MPX 5550/5650 appliances field replaceable?

No. The power supply on the NetScaler MPX 5500 and MPX 5550/5650 appliances is fixed.

Does the MPX 8005/8015/8200/8400/8600/8800 appliance ship with two power supplies?

No. The MPX 8005/8015/8200/8400/8600/8800 appliance supports dual power supplies but ships with one power supply. Contact your Citrix sales representative to order a second power supply.

How many power supplies are shipped with each platform?

The following table lists the number of power supplies shipped with each platform:

Platform	Number of Power Supplies shipped
MPX 5500	1

Platform	Number of Power Supplies shipped
MPX 7500/9500	1 (You can order a second power supply.)
MPX 9700/10500/12500/15500	2
MPX 15000/17000	1 (You can order a second power supply.)
MPX 11500/13500/14500/16500/18500/20500	2
MPX 17500/19500/21500	1 (You can order a second power supply.)
MPX 17550/19550/20550/21550	2

Are power supplies hot-swappable?

Yes. If the appliance has two power supplies, you can replace one power supply without shutting down the appliance, provided the other power supply is working.

Do you have different rail kits for 1U and 2U appliances?

No. All MPX and SDX appliances use the same rail kit. The kit contains two pairs of slide rails, of different lengths, for a 1U and a 2U appliance.

Which rail kit should I buy?

The appliance ships with the standard 4-post rail kit that fits racks from 28-38 inches.

The compact 4-post rail kit for racks from 23-33 inches, or the 2-post rail kit for 2-post racks, has to be purchased separately. Contact your Citrix sales representative to order the appropriate kit.

What are the maximum and the minimum lengths of the outer rack rails?

The length of a standard outer rack rail is from 28 to 38 inches. The length of a shorter outer rack rail is from 23 to 33 inches.

What is the space required between the front post and rear post of the rack?

Standard racks require 28–38 inches between the front and rear posts. Shorter racks require from 23 to 33 inches.

How far can an appliance extend from the front post of the rack?

The chassis can extend up to 1.25 inches from the front post for all NetScaler MPX and SDX appliances.

How much space is required for maintaining the front and rear area of an appliance?

Minimum clearance areas of 36 inches for the front area and 24 inches for the rear area are required for maintenance of all NetScaler MPX and SDX appliances.

Which LOM features are supported on the NetScaler MPX Appliance?

The MPX 8005/8015/8200/8400/8600/8800, MPX 11500/13500/14500/16500/18500/20500, and MPX

17550/19550/20550/21550 have an Intelligent Platform Management Interface (IPMI), also known as the Lights out Management (LOM) port, on the front panel of the appliance. The following three LOM features are supported on those platforms:

- Configuring the LOM port
- Power cycling the appliance
- Performing a core dump

Can the LOM interface be configured to accept only encrypted Virtual Network Computer (VNC) sessions on TCP port 5900?

Yes, customers who enable Transport Layer Security (TLS) on their LOM interface will have their VNC connections delivered over TLS as well.

For more information on LOM security guidelines, see Secure Deployment Guide for NetScaler MPX, VPX, and SDX Appliances.

Can the version of SSH used on the LOM interface be upgraded? Is there a patch available?

Individual components of the LOM cannot be upgraded independently. You must upgrade the entire LOM firmware as a package. The latest available LOM package can be found on the Citrix downloads website under LOM Firmware Upgrade.

Is it possible to add a third-party or self-signed SSL certificate to the LOM interface?

Yes, you can enable SSL on the latest binaries for third-party and self-signed SSL certificates, except on the 88XX models. On those models, the current LOM release does not support third-party certificates.

What is the recommended terminal emulator?

PuTTY.

Which platforms support Pay-As-You-Grow licenses?

The following platforms support Pay-As-You-Grow licenses:

- MPX 5550 to MPX 5650
- MPX 7500 to MPX 9500
- MPX 8005 to MPX 8015 to MPX 8200 to MPX 8400 to MPX 8600 to MPX 8800
- MPX 11500 to MPX 13500 to MPX 14500 to MPX 16500 to MPX 18500 to MPX 20500
- MPX 17500 to MPX 19500 to MPX 21500
- MPX 17550 to MPX 19550 to MPX 20550 to MPX 21550
- MPX 22040 to MPX 22060 to MPX 22080 to MPX 22100 to MPX 22120

Do you support direct attach cable (DAC)?

Yes, Citrix NetScaler appliances support a passive DAC in the following releases and builds:

- Release 9.3, build 63.4 and later
- Release 9.3.e, build 60.3007.e and later
- Release 10, build 74.2 and later
- Release 10.1, build 112.15 and later

Which port should I insert the DAC into?

DAC is inserted into the 10G port on the appliance.

Does the 1G port support DAC?

No. The DAC might fit into a 1G port but is not supported.

How can I order a DAC?

Contact your Citrix sales representative to order a DAC.

Can I mix DAC and fiber transceivers on the same appliance?

Yes. You can mix DAC and fiber transceivers on the same appliance. Each 10G port supports both options.

Can I mix SFP+ fiber and DAC in ports that are part of the same link aggregation channel (LAC)?

No. There must be symmetry between all elements in the same LAC.

Licensing, Upgrading, and Downgrading

Mar 17, 2012

The following topics describe the migration instructions for setting up a new version of a NetScaler with a list of all new and deprecated commands, parameters, and SNMP OIDs.

This document includes the following information:

- Changes in Release 10.1
- New and Deprecated Commands, Parameters, and SNMP OIDs
- NetScaler Licensing Overview
- NetScaler Gateway Universal License
- Upgrading or Downgrading the System Software
- Troubleshooting

- **Networking**
  Issue ID 0305772: The Use Source IP (USIP) parameter in an INAT rule is now disabled by default. In earlier releases, the USIP parameter was enabled by default.

Welcome to the NetScaler 10.1 system software release. There are new commands, parameters, and SNMP OIDs in this release. Some commands, parameters, and SNMP OIDs will be deprecated in this release. For complete descriptions of the new commands and parameters, see "Command Reference". For complete descriptions of the SNMP OIDs, see *— Citrix NetScaler SNMP OID Reference Guide* at "http://support.citrix.com/article/CTX137665".

New Commands

The following table lists all the new commands in release 10.1.

Table 1. New Commands

Command Group	Command
AAA	add aaa kcdAccount rm aaa kcdAccount set aaa kcdAccount show aaa kcdAccount count aaa kcdAccount
Authorization	rename authorization policy rename authorization policylabel

Command Group	Command
Cluster	add cluster nodegroup show cluster nodegroup set cluster nodegroup unset cluster nodegroup bind cluster nodegroup unbind cluster nodegroup rm cluster nodegroup
Content Optimization	show co policy bind co global unbind co global
Content Switching	rename cs policy rename cs policylabel
DB	add db dbProfile rm db dbProfile set db dbProfile unset db dbProfile show db dbProfile
Network	add nat64 set nat64 set rsskeytype show rsskeytype set L4Param show L4Param add trafficdomain bind trafficdomain unbind trafficdomain

Command Group	Command
	show trafficdomain rm trafficdomain
NetScaler	clear ns stats
QOS	show qos stats
Traffic Management	add tm samlSSOProfile rm tm samlSSOProfile set tm samlSSOProfile
VPN	add vpn samlSSOProfile rm vpn samlSSOProfile set vpn samlSSOProfile

New Parameters

The following table lists all the new parameters in release 10.1.

Table 2. New Parameters

Command Group	Command
Basic	add service [-td] add serviceGroup [-td]
AAA	set aaa ldapParams [-defaultAuthenticationGroup] add aaa kcdAccount [-kcdAccount] [-keytab] [-principle] rm aaa kcdAccount [-kcdAccount] set aaa kcdAccount [-kcdAccount] [-keytab] [-principle] unset aaa kcdAccount [-kcdAccount] show aaa kcdAccount [-kcdAccount] count aaa kcdAccount [-kcdAccount]
Application Firewall	import appfw customSettings [-merge] [-sha1] update appfw customSettings [-mergeDefault]

Command Group	Command
	set appfw settings [-signatureAutoUpdate] import appfw signatures [-merge] [-sha1] update appfw signatures [-mergeDefault]
Authentication	add authentication radiusAction [-defaultAuthenticationGroup] set authentication radiusAction [-defaultAuthenticationGroup] add authentication ldapAction [-defaultAuthenticationGroup] set authentication ldapAction [-defaultAuthenticationGroup] add authentication tacacsAction [-defaultAuthenticationGroup] set authentication tacacsAction [-defaultAuthenticationGroup] add authentication negotiateAction [-defaultAuthenticationGroup] set authentication negotiateAction [-defaultAuthenticationGroup] add authentication samlAction [-defaultAuthenticationGroup] set authentication samlAction [-defaultAuthenticationGroup] add authentication vserver [-td] [-ownerNode] set authentication vserver [-ngname] add authentication authnProfile [-Authentication] [-authnVsName] [-AuthenticationHost] [-AuthenticationDomain] [-AuthenticationLevel] set authentication authnProfile [-Authentication] [-authnVsName] [-AuthenticationHost] [-AuthenticationDomain] [-AuthenticationLevel]
Authorization	rename authorization policy [-name] [-newName] rename authorization policylabel [-labelName] [-newName]
Cache	add cache contentGroup [-persist] set cache contentGroup [-persist] flush cache contentGroup [-force] flush cache object [-force]
Cluster	add cluster node [-priority] set cluster node [-priority]

Command Group	Command
	unset cluster node [-priority]
Content Optimization	bind co global [-policyName] unbind co global [-policyName]
Cache Redirection	add cr vserver [-td] set cr vserver [-ngname] bind cr vserver [-lbvserver] unbind cr vserver [-lbvserver]
Content Switching	add cs policy [-logAction] set cs policy [-logAction] unset cs policy [-logAction] rename cs policy [-policyName] [-newName] rename cs policylabel [-labelName] [-newName] add cs vserver [-td] [-soBackupAction] [-dbProfileName] [-oracleServerVersion] set cs vserver [-soBackupAction] [-dbProfileName] [-oracleServerVersion] unset cs vserver [-dbProfileName] [-authnProfile] add cs action [-targetVserverExpr] set cs action [-targetVserverExpr]
DB	add db dbProfile [-name] [-interpretQuery] [-stickiness] [-kcdAccount] rm db dbProfile [-name] set db dbProfile [-name] [-interpretQuery] [-stickiness] [-kcdAccount] unset db dbProfile [-name] [-interpretQuery] [-stickiness] [-kcdAccount] show db dbProfile [-name]
Global Server Load Balancing	add gslb vserver [-soBackupAction] set gslb vserver [-soBackupAction] bind gslb vserver [-policyName] unbind gslb vserver [-policyName]

Command Group	Command
HA	set HA node [-maxFlips] [-maxFlipTime] [-syncvlan]
Network	add ipsec profile [-ikeVersion] set ipsec parameter [-ikeVersion] unset ipsec parameter [-ikeVersion] add arp [-td] rm arp [-td] add route [-td] rm route [-td] set route [-td] unset route [-td] add vlan [-sdxVlan] set vlan [-sdxVlan] add route6 [-td] rm route6 [-td] set route6 [-td] unset route6 [-td] add nd6 [-td] rm nd6 [-td] add inat [-td] add inat -mode STATELESS [-tftp (ENABLED | DISABLED)] set inat -mode STATELESS [-tftp (ENABLED | DISABLED)] set inatparam [-nat46IgnoreTOS (YES | NO)] [-nat46ZeroCheckSum (ENABLED | DISABLED)] [-nat46v6Mtu <positive_integer>][-nat46FragHeader (ENABLED | DISABLED)] add netProfile [-td] clear rnat [-td] set lacp [-ownerNode] show lacp [-ownerNode] rm lacp [-ownerNode]

Command Group	Command
	set ipv6 [-routerRedirection] [-doDAD] set inatparam [-nat46v6Prefix]
Load Balancing	add lb monitor [-storename] [-storefrontacctservice] [-hostName] [-kcdAccount] set lb monitor [-storename] [-storefrontacctservice] [-hostName] [-kcdAccount] add lb vserver [-soBackupAction] [-dbProfileName] [-oracleServerVersion] [-minAutoscaleMembers] [-maxAutoscaleMembers] [-skippersistency] [-td] set lb vserver [-soBackupAction] [-dbProfileName] [-oracleServerVersion] [-minAutoscaleMembers] [-maxAutoscaleMembers] [-skippersistency] unset lb vserver [-dbProfileName] [-skippersistency] [-minAutoscaleMembers] [-maxAutoscaleMembers] [-authnProfile] clear lb persistentSessions [-persistenceParameter]
NetScaler	set ns limitIdentifier [-ngname] add ns acl [-td] add ns acl6 [-td] set ns config -nwfwmode <nwfwmode> add ns ip6 [-td] rm ns ip6 [-td] add ns ip [-td] rm ns ip [-td] add ns simpleacl [-td] add ns simpleacl6 [-td] add ns pbr [-td] [-ipTunnel] set ns pbr [-ipTunnel] unset ns pbr [-ipTunnel] add ns tcpProfile [-sendBuffsize] set ns tcpProfile [-sendBuffsize] unset ns tcpProfile [-sendBuffsize] add ns httpProfile [-spdy]

Command Group	Command
	set ns httpProfile [-spdy] unset ns httpProfile [-spdy]
	set ns param [-useproxyport] [-internaluserlogin] add ns pbr6 [-td]
Policy	set policy httpCallout [-scheme] [-cacheForSecs] unset policy httpCallout [-cacheForSecs]
SNMP	set snmp engineId [-ownerNode] unset snmp engineId [-ownerNode] show snmp engineId [-ownerNode] rm snmp engineId [-ownerNode]
Stream	add stream selector [-builtin] add stream identifier [-ngname] [-builtin] set stream identifier [-ngname] [-builtin]
Traffic Management	add tm sessionAction [-kcdAccount] [-homePage] set tm sessionAction [-kcdAccount] [-homePage] add tm samlSSOProfile [-name] [-samlSigningCertName] [-assertionConsumerServiceURL] [-relaystateRule] [-sendPassword] [-samlIssuerName] rm tm samlSSOProfile [-name] set tm samlSSOProfile [-name] [-samlSigningCertName] [-assertionConsumerServiceURL] [-sendPassword] [-samlIssuerName] [-relaystateRule] show tm samlSSOProfile [-name] add tm trafficAction [-kcdAccount] [-samlSSOProfile] set tm trafficAction [-kcdAccount] [-samlSSOProfile] set tm sessionParameter [-kcdAccount] [-homePage] unset tm sessionParameter [-kcdAccount] [-homePage]
Utility	traceroute6 [-T] traceroute [-T]

Command Group	Command
	ping6 [-T] ping [-T]
VPN	add vpn vserver [-netProfile] [-ownerNode] set vpn vserver [-netProfile] bind vpn vserver [-appController] [-sharefile] unbind vpn vserver [-appController] [-sharefile] add vpn trafficAction [-kcdAccount] [-samlSSOProfile] set vpn trafficAction [-kcdAccount] [-samlSSOProfile] add vpn samlSSOProfile [-name] [-samlSigningCertName] [-assertionConsumerServiceURL] [-relaystateRule] [-sendPassword] [-samlIssuerName] rm vpn samlSSOProfile [-name] set vpn samlSSOProfile [-name] [-samlSigningCertName] [-assertionConsumerServiceURL] [-sendPassword] [-samlIssuerName] [-relaystateRule] show vpn samlSSOProfile [-name] add vpn sessionAction [-kcdAccount] set vpn sessionAction [-kcdAccount] bind vpn global [-appController] [-sharefile] unbind vpn global [-appController] [-sharefile] set vpn parameter [-mdxTokenTimeout] [-UITHEME] [-kcdAccount] unset vpn parameter [-mdxTokenTimeout] [-UITHEME] [-kcdAccount]
Web Interface	add wi site [-appWelcomeMessage] [-welcomeMessage] [-footerText] [-loginSysMessage] [-preLoginButton] [-preLoginMessage] [-preLoginTitle] [-domainSelection] [-userInterfaceBranding] [-ShowSearch] [-ShowRefresh] [-wiUserInterfaceModes] [-UserInterfaceLayouts] set wi site [-appWelcomeMessage] [-welcomeMessage] [-footerText] [-loginSysMessage] [-preLoginButton] [-preLoginMessage] [-preLoginTitle] [-domainSelection] [-userInterfaceBranding] [-ShowSearch] [-ShowRefresh] [-wiUserInterfaceModes] [-UserInterfaceLayouts]

Deprecated Parameters

The following parameters are deprecated in release 10.1.

- add cr vserver -cacheVserver
- set cr vserver -cacheVserver
- unset cr vserver -cacheVserver
- set system collectionparam -communityName

New SNMP OIDs

The following table lists the new SNMP OIDs in release 10.1.

OID		Description
ACL		
aclTotCount	1.3.6.1.4.1.5951.4.1.1.22.1.21	Total number of ACL rules configured
acl6TotPktsNAT64	1.3.6.1.4.1.5951.4.1.1.22.4.27	Packets matching a NAT64 ACL6, resulting in a NAT64 translation
acl6TotCount	1.3.6.1.4.1.5951.4.1.1.22.4.28	Total number of ACL6 rules configured
SPDY		
spdy2TotStreams	1.3.6.1.4.1.5951.4.1.1.48.72	Total number of requests received over SPDY
Cache		
cacheNumObjSavedOnDisk	1.3.6.1.4.1.5951.4.1.1.49.77	Cached responses currently saved on disk. Includes responses fully saved to disk, and expired or flushed but not yet removed
cacheNumMBSavedOnDisk	1.3.6.1.4.1.5951.4.1.1.49.78	Size (MB) of cached responses currently saved on disk. Includes responses fully saved to disk, and expired or flushed but not yet removed
cacheNumMBReadFromDisk	1.3.6.1.4.1.5951.4.1.1.49.79	Total Number of bytes read from disk since last reboot
cacheNumMBWrittenToDisk	1.3.6.1.4.1.5951.4.1.1.49.80	Total Number of MB written to disk since last reboot
SSL-VPN		
iPv6tov4FindIPv6MapErr	1.3.6.1.4.1.5951.4.1.1.66.44	Number of IPv6toIPv4 find IPv6 mapping errors
iPv6toV4MapInsertErr	1.3.6.1.4.1.5951.4.1.1.66.45	Number of Ipv6 to Ipv4 mapping Insert Errors
parseIPv6AddressErr	1.3.6.1.4.1.5951.4.1.1.66.46	Errors in parsing for Ipv6 address from address string
AAA		

OID		Description
aaaCurTMSessions	1.3.6.1.4.1.5951.4.1.1.67.11	Count of current AAATM sessions
aaaTotTMSessions	1.3.6.1.4.1.5951.4.1.1.67.12	Count of all AAATM sessions
INAT Sessions		
nsInatGlobalStats	1.3.6.1.4.1.5951.4.1.1.74.1	This provides statistics related to all INAT sessions
nat46TotTcp46	1.3.6.1.4.1.5951.4.1.1.74.1.1	Total TCP packets translated (V4->v6)
nat46TotUdp46	1.3.6.1.4.1.5951.4.1.1.74.1.2	Total UDP packets translated (V4->v6)
nat46TotIcmp46	1.3.6.1.4.1.5951.4.1.1.74.1.3	Total ICMP packets translated (V4->v6)
nat46TotDrop46	1.3.6.1.4.1.5951.4.1.1.74.1.4	Total IPV4 packets dropped
nat46TotTcp64	1.3.6.1.4.1.5951.4.1.1.74.1.5	Total TCP packets translated (V6->v4)
nat46TotUdp64	1.3.6.1.4.1.5951.4.1.1.74.1.6	Total UDP packets translated (V6->v4)
nat46TotIcmp64	1.3.6.1.4.1.5951.4.1.1.74.1.7	Total ICMP packets translated (V6->v4)
nat46TotDrop64	1.3.6.1.4.1.5951.4.1.1.74.1.8	Total IPV6 packets dropped
nsInatPerNat46StatsTable	1.3.6.1.4.1.5951.4.1.1.74.2	This provides statistics related to per nat46 rule
nsInatPerNat46StatsEntry	1.3.6.1.4.1.5951.4.1.1.74.2.1	
inatname	1.3.6.1.4.1.5951.4.1.1.74.2.1.1	The name of the INAT
inatNat46Tcp46	1.3.6.1.4.1.5951.4.1.1.74.2.1.2	TCP packets translated (V4->v6)
inatNat46Udp46	1.3.6.1.4.1.5951.4.1.1.74.2.1.3	UDP packets translated (V4->v6)
inatNat46Icmp46	1.3.6.1.4.1.5951.4.1.1.74.2.1.4	ICMP packets translated (V4->v6)
inatNat46Drop46	1.3.6.1.4.1.5951.4.1.1.74.2.1.5	IPV4 packets dropped

OID		Description
inatNat46Tcp64	1.3.6.1.4.1.5951.4.1.1.74.2.1.6	TCP packets translated (V6->v4)
inatNat46Udp64	1.3.6.1.4.1.5951.4.1.1.74.2.1.7	UDP packets translated (V6->v4)
inatNat46Icmp64	1.3.6.1.4.1.5951.4.1.1.74.2.1.8	ICMP packets translated (V6->v4)
inatNat46Drop64	1.3.6.1.4.1.5951.4.1.1.74.2.1.9	IPV6 packets dropped
nsInatPerNatStatsTable	1.3.6.1.4.1.5951.4.1.1.74.3	This provides statistics related to per inat session
nsInatPerNatStatsEntry	1.3.6.1.4.1.5951.4.1.1.74.3.1	
inat44name	1.3.6.1.4.1.5951.4.1.1.74.3.1.1	The name of the INAT
inatTotHits	1.3.6.1.4.1.5951.4.1.1.74.3.1.2	INAT total sessions
inatCurSessions	1.3.6.1.4.1.5951.4.1.1.74.3.1.3	INAT current sessions
inatTotReceiveBytes	1.3.6.1.4.1.5951.4.1.1.74.3.1.4	INAT total Received Bytes
inatTotSentBytes	1.3.6.1.4.1.5951.4.1.1.74.3.1.5	INAT total Sent Bytes
inatTotpktreceived	1.3.6.1.4.1.5951.4.1.1.74.3.1.6	INAT total Packets Received
inatTotpktsent	1.3.6.1.4.1.5951.4.1.1.74.3.1.7	INAT total Packets Sent
NAT64		
nsNat64GlobalStats	1.3.6.1.4.1.5951.4.1.1.75.1	This provides statistics related to all NAT64 sessions
nat64TotUdpSessions	1.3.6.1.4.1.5951.4.1.1.75.1.1	Total number of UDP sessions created by NAT64
nat64TotTcpSessions	1.3.6.1.4.1.5951.4.1.1.75.1.2	Total number of TCP sessions created by NAT64
nat64TotSessions	1.3.6.1.4.1.5951.4.1.1.75.1.3	Total number of sessions created by NAT64
nat64TotIcmpSessions	1.3.6.1.4.1.5951.4.1.1.75.1.4	Total number of ICMP sessions created by NAT64
IPv6		

OID		Description
nsIp6StatsGroup	1.3.6.1.4.1.5951.4.1.1.76	This provides information about IPv6 related statistics in the NetScaler product
ipv6TotRxPkts	1.3.6.1.4.1.5951.4.1.1.76.1	IPv6 packets received
ipv6TotTxPkts	1.3.6.1.4.1.5951.4.1.1.76.2	IPv6 packets transmitted
ipv6TotRxBytes	1.3.6.1.4.1.5951.4.1.1.76.3	Bytes of IPv6 data received
ipv6TotTxBytes	1.3.6.1.4.1.5951.4.1.1.76.4	Bytes of IPv6 data transmitted
ipv6FragTotRxPkts	1.3.6.1.4.1.5951.4.1.1.76.5	IPv6 fragments received
ipv6FragTotPktsForward	1.3.6.1.4.1.5951.4.1.1.76.6	IPv6 fragments forwarded to the client or server without reassembly
ipv6FragTotPktsProcessNoReass	1.3.6.1.4.1.5951.4.1.1.76.7	IPv6 fragments processed without reassembly
ipv6ErrHdr	1.3.6.1.4.1.5951.4.1.1.76.8	Packets received that contain an error in one or more components of the IPv6 header
ipv6LandAttack	1.3.6.1.4.1.5951.4.1.1.76.9	Land-attack packets received. The source and destination addresses are the same. If not dropped, these packets can lock up the appliance
ipv6FragZeroLenPkt	1.3.6.1.4.1.5951.4.1.1.76.10	Packets received with a fragment length of 0 bytes
ipv6TotIcmpFragPkts	1.3.6.1.4.1.5951.4.1.1.76.11	Number of ICMPV6 fragmented packets
ipv6TotLookupDone	1.3.6.1.4.1.5951.4.1.1.76.12	Total number of nd6 lookup done
ipv6TotLookupFailed	1.3.6.1.4.1.5951.4.1.1.76.13	Total number of nd6 lookup failed
ipv6TotStaticRoutes	1.3.6.1.4.1.5951.4.1.1.76.14	Total number of static ipv6 routes
ipv6TotDynamicRoutes	1.3.6.1.4.1.5951.4.1.1.76.15	Total number of dynamic ipv6 routes
ipv6TotNeigborDiscovered	1.3.6.1.4.1.5951.4.1.1.76.16	Total number of nd6 entries both dynamic and static
ipv6TotIpv6To4Conversions	1.3.6.1.4.1.5951.4.1.1.76.17	Total number of ipv6 to v4 conversion done

OID		Description
ipv6TotIpv4To6Conversions	1.3.6.1.4.1.5951.4.1.1.76.18	Total number of ipv4 to v6 conversion done
ipv6TotTcpConnection	1.3.6.1.4.1.5951.4.1.1.76.19	TCP connections over IPv6
ipv6TotNonTcpConnection	1.3.6.1.4.1.5951.4.1.1.76.20	Non TCP connections over IPv6
Traffic Domain		
nsTdInetAddressTable	1.3.6.1.4.1.5951.4.1.1.77	This table contains information about the non-default Td IP Addresses configured on the NetScaler
nsTdInetAddressEntry	1.3.6.1.4.1.5951.4.1.1.77.1	
nsTdInetId	1.3.6.1.4.1.5951.4.1.1.77.1.1	This represents a traffic domain ID
nsTdInetAddressType	1.3.6.1.4.1.5951.4.1.1.77.1.2	The address type of nsTdInetAddress
nsTdInetAddress	1.3.6.1.4.1.5951.4.1.1.77.1.3	This represents an IPv4/v6 address configured on the NetScaler
nsTdInetMaskLength	1.3.6.1.4.1.5951.4.1.1.77.1.4	This represents netmask length
nsTdInetType	1.3.6.1.4.1.5951.4.1.1.77.1.5	This represents the IP address type
nsTdInetMode	1.3.6.1.4.1.5951.4.1.1.77.1.6	This represents the IP address mode
nsTdInetFreePorts	1.3.6.1.4.1.5951.4.1.1.77.1.7	This represents the number of unused ports free on this IP
nsTdInetVlan	1.3.6.1.4.1.5951.4.1.1.77.1.8	The vlan to which this ip address is bound
nsTdInetBridgeGroup	1.3.6.1.4.1.5951.4.1.1.77.1.9	The bridge group to which this ip address is bound
svcTdId	1.3.6.1.4.1.5951.4.1.2.1.1.57	Traffic Domain ID of this service
serverTdId	1.3.6.1.4.1.5951.4.1.2.2.1.8	Traffic Domain ID of this server
svcGrpMemberTdId	1.3.6.1.4.1.5951.4.1.2.7.1.37	Traffic Domain ID of this service group member
svcgrpTdId	1.3.6.1.4.1.5951.4.1.2.11.1.5	Traffic Domain ID of this service group

OID		Description
vsvrTdId	1.3.6.1.4.1.5951.4.1.3.1.1.69	Traffic Domain of the vserver
Load Balancing		
curConfigLbVservers	1.3.6.1.4.1.5951.4.1.3.5.4	Total number of LB vservers configured on the NetScaler
curConfigGslbVservers	1.3.6.1.4.1.5951.4.1.3.5.5	Total number of GSLB vservers configured on the NetScaler
totSpilloverCount	1.3.6.1.4.1.5951.4.1.3.5.6	Total count of spillovers
SNMP Trap		
qosdVersion	1.3.6.1.4.1.5951.4.1.10.2.39	This counter tells the QOSD version
brVersion	1.3.6.1.4.1.5951.4.1.10.2.40	This counter tells the BR version

NetScaler Licensing Overview

Aug 24, 2016

If you want to upgrade your software to 10.1, you can use your existing license. Contact your Citrix sales representative for new licenses if you are using the standard edition and want to upgrade to the enterprise or platinum edition, or if you are using the enterprise edition and want to upgrade to the platinum edition.

You can easily allocate your NetScaler licenses. In the NetScaler configuration utility (GUI), you can use your hardware serial number (HSN) or your license activation code (LAC) to allocate your licenses. Alternatively, if a license is already present on your local computer, you can upload it to the appliance.

For all other functionality, such as returning or reallocating your license, you must use the licensing portal. Optionally, you can still use the licensing portal for license allocation. For more information about the licensing portal, see "http://support.citrix.com/article/CTX131110".

Note:

- On a NetScaler MPX or SDX appliance, you can use the HSN or LAC to allocate your license or upload the license to the appliance from a local computer. On a NetScaler VPX appliance, you can only upload the license to the appliance from a local computer.
- You must purchase separate licenses for each appliance in a high availability (HA) pair. Make sure that the same type of licenses are installed on both the appliances. For example, if you purchase a platinum license for one appliance, you must purchase another platinum license for the other appliance.

This document includes the following information:

Updated: 2014-06-24

To use the hardware serial number or license activation code to allocate your licenses:

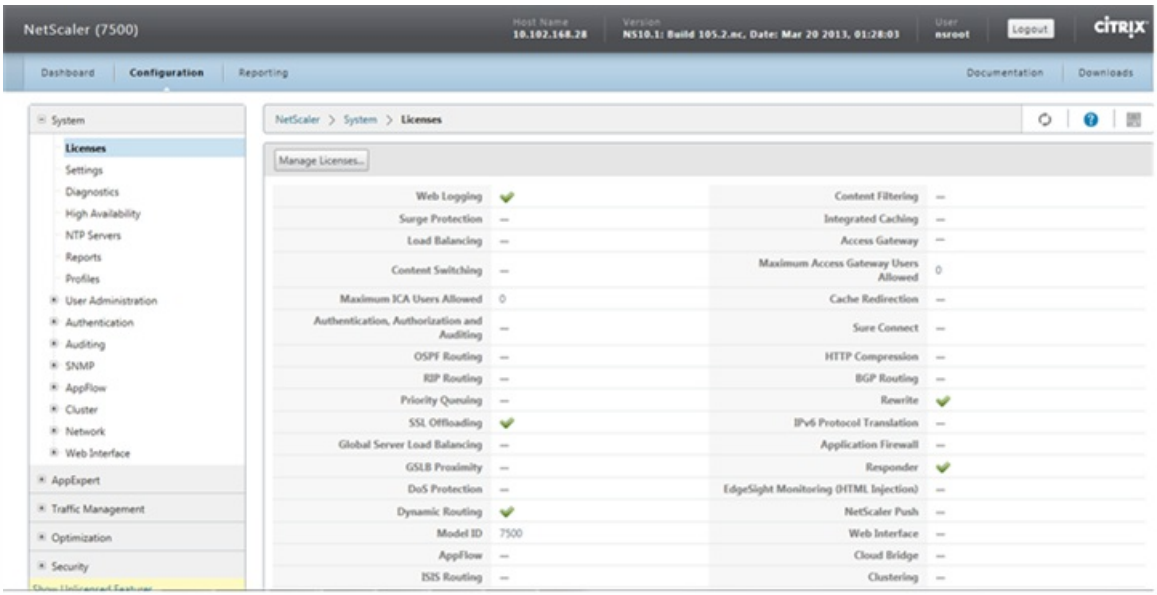
- You must be able to access public domains through the appliance. The license allocation software internally accesses the Citrix licensing portal for your license. To access a public domain, you must configure a NetScaler IP (NSIP) address, configure a mapped IP (MIP) address or a subnet IP (SNIP) address, and set up a DNS server.
- Your license must be linked to your hardware, or you must have a valid license activation code (LAC). Citrix sends your LAC by email when you purchase a license.

Updated: 2014-02-11

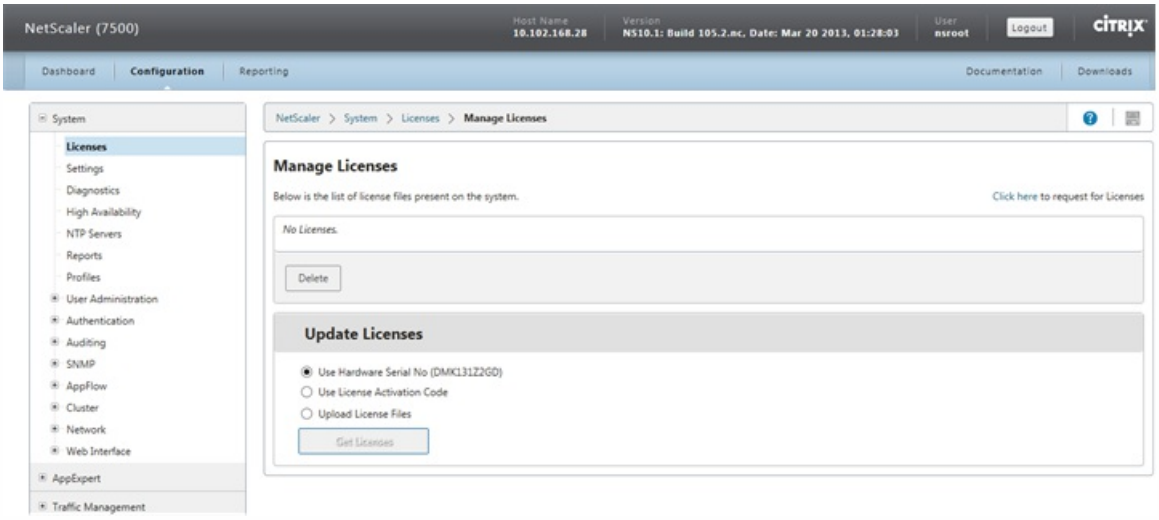
If your license is already linked to your hardware, the license allocation process can use the hardware serial number. Otherwise, you must type the license activation code (LAC).

To allocate your license

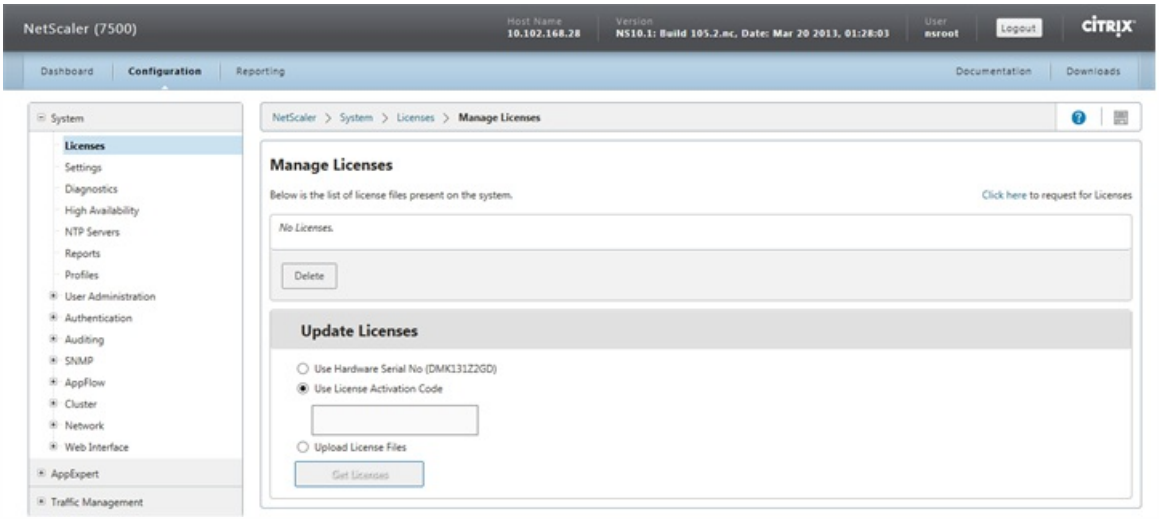
1. In a web browser, type the IP address of the NetScaler (for example, http://192.168.100.1).
2. In User Name and Password, type the administrator credentials.
3. On the Configuration tab, navigate to System > Licenses.
4. In the details pane, click Manage Licenses.



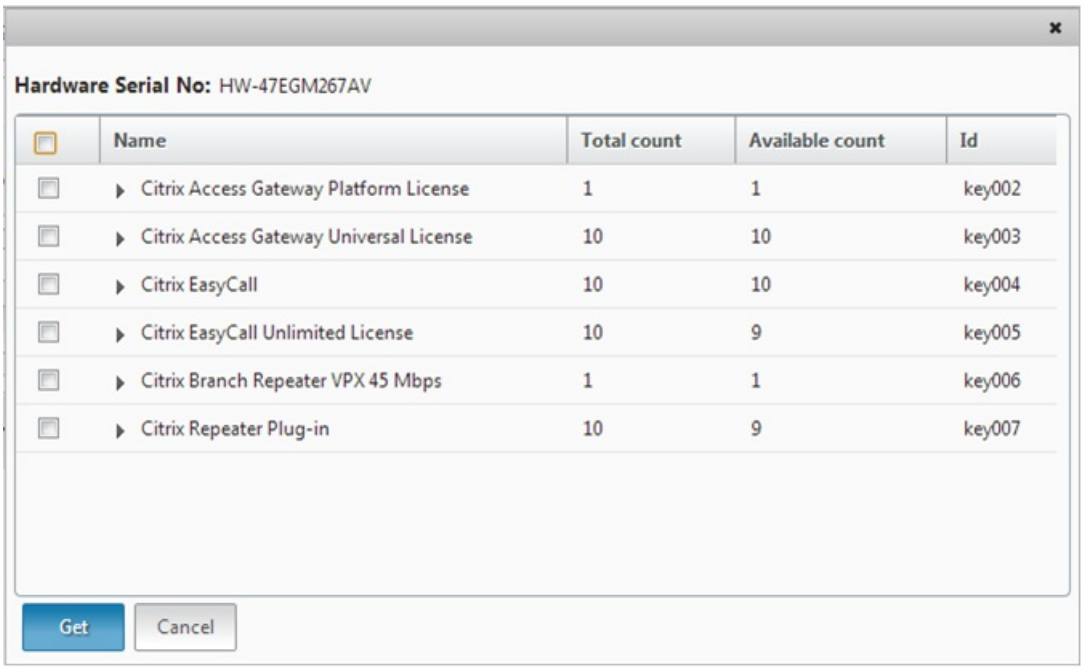
5. Click Update Licenses, and then select one of the following options:
   - **Use Hardware Serial Number**—The software internally fetches the serial number of your appliance and uses this number to display your license(s).



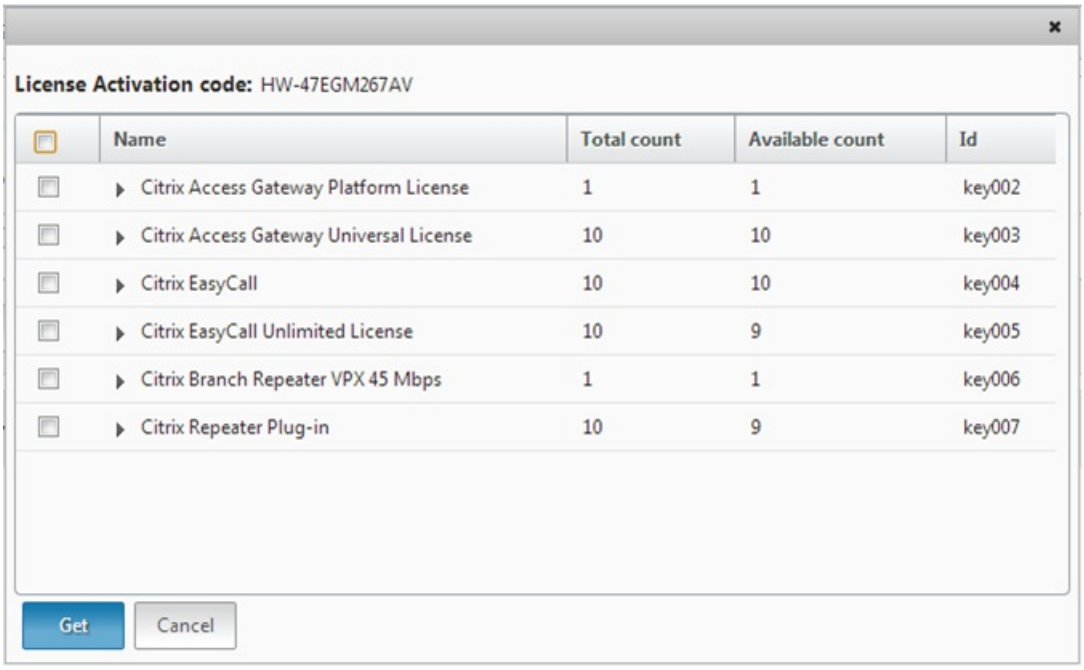
   - **Use License Activation Code**—Citrix emails the LAC for the license that you purchased. Enter the LAC in the text box.



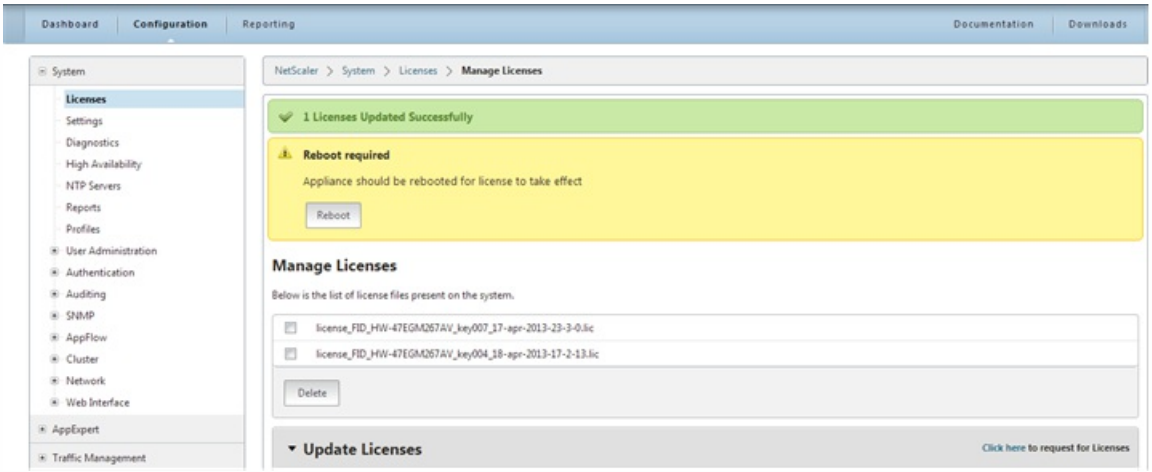
6. Click Get Licenses. Depending on the option that you selected, one of the following dialog boxes appears.
   - The following dialog box appears if you selected Hardware Serial Number.



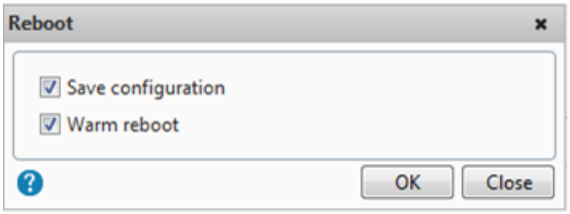
   - The following dialog box appears if you selected License Activation Code.



7. Select the license that you want to allocate, and then click Get.
8. Click Reboot for the license to take effect.



9. In the Reboot dialog box, click OK to proceed with the changes, or click Close to cancel the changes.

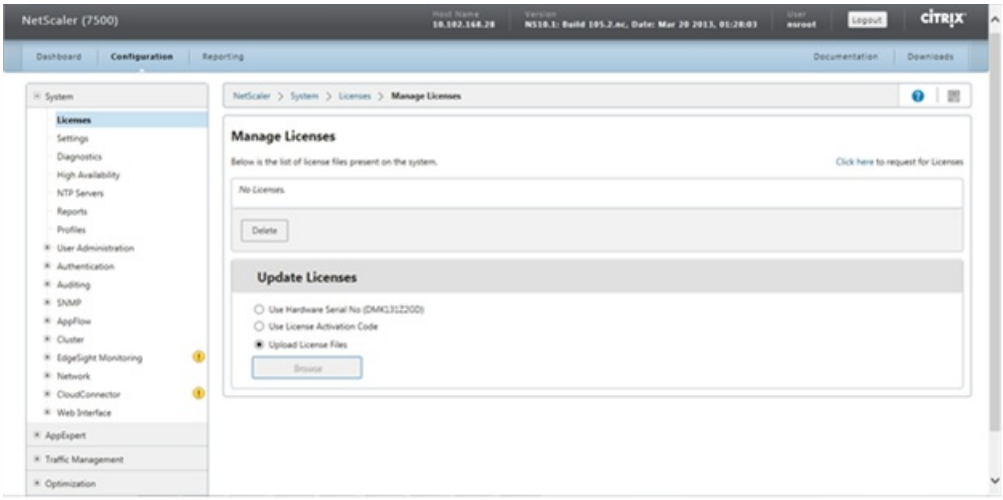


Updated: 2014-06-24

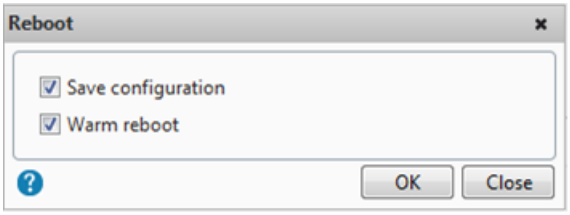
If you downloaded your license file to your local computer by accessing the licensing portal, you must upload the license to the appliance.

To install a license file by using the configuration utility

1. In a web browser, type the IP address of the NetScaler (for example, http://192.168.100.1).
2. In User Name and Password, type the administrator credentials.
3. On the Configuration tab, navigate to System > Licenses .
4. In the details pane, click Manage Licenses.
5. Click Update Licenses, and then select **Upload License Files**.



6. Click Browse. Navigate to the location of the license files, select the license file, and then click Open.
7. Click Reboot to apply the license.
8. In the Reboot dialog box, click OK to proceed with the changes, or click Close to cancel the changes.



See also

To install the licenses by using the command line interface

1. Open an SSH connection to the NetScaler by using an SSH client, such as PuTTY.
2. Log on to the NetScaler by using the administrator credentials.
3. Switch to the shell prompt, create a license subdirectory in the nsconfig directory, if it does not exist, and copy the new license file(s) to this directory.

Example

```
login: nsroot
Password: nsroot
Last login: Mon Aug  4 03:37:27 2008 from 10.102.29.9
Done
> shell
Last login: Mon Aug  4 03:51:42 from 10.103.25.64
root@ns# mkdir /nsconfig/license
root@ns# cd /nsconfig/license
```

Copy the new license file(s) to this directory.

Note: The NetScaler appliance does not prompt for a reboot option when you use the command line interface to install the licenses. Run the reboot -w command to warm reboot the system, or run the reboot command to reboot the system normally.

Updated: 2014-06-24

Before using a feature, make sure that your license supports the feature.

To verify the licensed features by using the command line interface

1. Open an SSH connection to the NetScaler by using an SSH client, such as PuTTY.
2. Log on to the NetScaler by using the administrator credentials.
3. At the command prompt, enter the sh ns license command to display the features supported by the license.

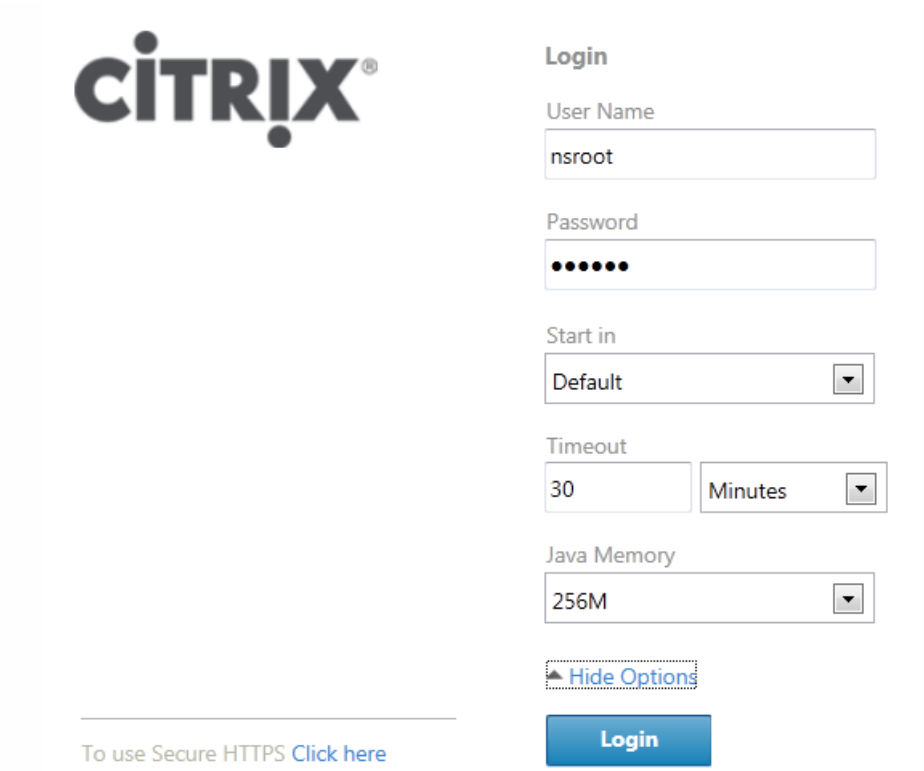
Example

```
sh ns license
        License status:
                    Web Logging: YES
                Surge Protection: YES
                  .......

                  HTML Injection: YES
 Done
```

To verify the licensed features by using the configuration utility

1. In a Web browser, type the IP address of the NetScaler, such as http://192.168.100.1.
2. In User Name and Password, type the administrator credentials.
3. In Start in, select Configuration, and then click Login, as shown in the following figure.

Figure 1. Login Screen



4. In the navigation pane, expand System, and then click Licenses. You will see a green check mark next to the licensed features.

Updated: 2014-07-01

When you use the NetScaler appliance for the first time, you need to enable a feature before you can use its functionality. If you configure a feature before it is enabled, a warning message appears. The configuration is saved but it will apply only after the feature is enabled.

To enable a feature by using the command line interface

At the NetScaler command prompt, type the following commands to enable a feature and verify the configuration:

- enable feature <FeatureName>
- show feature

Example

```
enable feature lb cs
done
 >show feature

      Feature                Acronym          Status
      -------                -------          ------
 1)   Web Logging             WL               OFF
 2)   Surge Protection        SP               ON
 3)   Load Balancing          LB               ON
 4)   Content Switching       CS               ON
 5)   Cache Redirection       CR               ON
```



```
.
.
.
24)  NetScaler Push          push          OFF
Done
```

The example shows how to enable load balancing (lb) and content switching (cs).

If the license key is not available for a particular feature, the following error message appears for that feature:

```
ERROR: feature(s) not licensed
```

Note: To enable an optional feature, you need a feature-specific license. For example, if you have purchased and installed the Citrix NetScaler Enterprise Edition license and need to enable the Integrated Caching feature, you first need to purchase and install the AppCache license.

To disable a feature by using the command line interface

At the NetScaler command prompt, type the following commands to disable a feature and verify the configuration:

- disable feature <FeatureName>
- show feature

Example

The following example shows how to disable load balancing (LB).

```
> disable feature lb
Done
> show feature

     Feature              Acronym        Status
     -------              -------        ------
1)   Web Logging          WL             OFF
2)   Surge Protection     SP             ON
3)   Load Balancing       LB             OFF
4)   Content Switching    CS             ON
.
.
.
24)  NetScaler Push       push           OFF
Done
>
```

Licensing

Jul 25, 2013

Before you can deploy Citrix NetScaler Gateway to support user connections, the appliance must be properly licensed.

Important: Citrix recommends that you retain a local copy of all license files you receive. When you save a backup copy of the configuration file, all uploaded licenses files are included in the backup. If you need to reinstall NetScaler Gateway appliance software and do not have a backup of the configuration, you will need the original license files.
Before installing licenses on NetScaler Gateway, set the host name of the appliance and then restart NetScaler Gateway. You use the Setup Wizard to configure the host name. When you generate the Universal license for NetScaler Gateway, the host name is used in the license.

Upgrading or Downgrading the System Software

Jun 25, 2014

NetScaler 10.1 offers new and updated features with increased functionality. A comprehensive list of enhancements is listed in the release notes accompanying the release announcement. Take a moment to read this document before you upgrade your software.

It is important to understand the licensing framework and types of licenses before you upgrade your software. A software edition upgrade may require new licenses, such as upgrading from the standard edition to the enterprise edition, the standard edition to the platinum edition, or the enterprise edition to the platinum edition.

Note: For upgrading or downgrading the nodes in a cluster setup, see "Upgrading or Downgrading the Cluster Software". Upgrading from release 10.1 build 121.10 or any earlier releases to release 10.1 build 122.17 and later involves some location changes of user monitor script files. For details, see Directory Locations of Script Files for User Monitors.

This document includes the following information:

- Upgrading to Release 10.1
- Upgrading to a Later Build within Release 10.1
- Downgrading from Release 10.1
- Downgrading to an Earlier Build within Release 10.1
- Auto Cleanup

Updated: 2014-06-24

In release 10.1 build 122.17, the script files for user monitors are at a new location. If you upgrade an appliance or virtual appliance to release 10.1 build 122.17 or later, the changes are as follows:

- A new directory named conflicts is created in /nsconfig/monitors/ and all the built-in scripts of the previous builds are moved to this directory.
- All new built-in scripts are available in the /netscaler/monitors/ directory. All custom scripts are available in the /nsconfig/monitors/ directory.
- You must save a new custom script in the /nsconfig/monitors/ directory.
- After the upgrade is completed, if a custom script is created and saved in the /nsconfig/monitors/ directory with the same name as that of a built-in script, the script in the /netscaler/monitors/ directory takes priority. That is, the custom script is not run.

If you provision a virtual appliance running release 10.1 build 122.17 or later, the changes are as follows:

- All built-in scripts are available in the /netscaler/monitors/ directory
- The directory /nsconfig/monitors/ is empty.
- If you create a new custom script, you must save it in the /nsconfig/monitors/ directory.

For more information about user monitors, see "Understanding User Monitors."

Upgrading to Release 10.1

Dec 30, 2015

You can use the configuration utility to upgrade most older releases to 10.1. For others, you must use the command line interface. You follow the same basic procedure to upgrade either a standalone appliance or each appliance in a high availability pair, although additional considerations apply to upgrading a high availability pair.

This document includes the following information:

- Upgrading a Standalone NetScaler
- Upgrading a High Availability Pair

Updated: 2014-12-10

Before upgrading the system software, make sure that you have the required licenses. For more information, see "NetScaler Licensing Overview." You do not need a new license for the following upgrades:

- 8.x to 9.x
- 8.x or 9.x to 10.x
- 8.x, 9.x, or 10.x to 10.y

Note: When upgrading from release 8.0, 8.1, 9.0, 9.1, 9.2, 9.3, or 10, you have the option to use the configuration utility or the command line interface. Citrix recommends to perform the upgrade by using the command line interface. When using the upgrade wizard in the configuration utility to upgrade from release 8.0, do not use the Device option to upload your software.

In the following procedure, <release> and <releasenumber> represent the release version you are upgrading to, and <targetbuildnumber> represents the build number that you are upgrading to. The procedure includes optional steps to avoid losing any updates that are pushed to the /etc directory during the upgrade.

To upgrade a standalone NetScaler appliance running release 8.0, 8.1, 9.0, 9.1, 9.2, 9.3, or 10 by using the command line interface

1. Use an SSH client, such as PuTTy, to open an SSH connection to the appliance.
2. Log on to the appliance by using the administrator credentials. Save the running configuration. At the prompt, type:save config
3. Create a copy of the ns.conf file. At the shell prompt, type:
   1. cd /nsconfig
   2. cp ns.conf ns.conf.NS<currentreleasenumber><currentbuildnumber>

   You should backup the configuration file to another computer.
4. (Optional) If you have modified some of the following files in the /etc directory, and copied them to /nsconfig to maintain persistency, any updates that are pushed to the /etc directory during the upgrade might be lost:
   - ttys
   - resolv.conf
   - sshd_config
   - host.conf
   - newsyslog.conf
   - host.conf

- httpd.conf
- rc.conf
- syslog.conf
- crontab
- monitrc

To avoid losing these updates, create a /var/nsconfig_backup directory, and move the customized files to this directory. That is, move any files that you modified in /etc directory and copied to /nsconfig by running the following command:

cp /nsconfig/<filename> /var/nsconfig_backup

Example:
cp /nsconfig/syslog.conf /var/nsconfig_backup

5. Create a location for the installation package. At the shell prompt type
   1. cd/var/nsinstall
   2. mkdir <releasenumber>nsinstall
   3. cd <releasenumber>nsinstall
   4. mkdir build_<targetbuildnumber>
   5. cd build_<targetbuildnumber>
6. Download or copy the installation package (build-<release>-<targetbuildnumber>_nc.tgz) to the directory that you created for it. To download the installation package from the Citrix Web site, do the following:
   1. Go to MyCitrix.com, log on with your credentials, and click Downloads.
   2. In Select a Product, select NetScaler ADC.
   3. Under Firmware, click the release and build number to download.
   4. Click Get Firmware.

   Note: Documentation is not available as part of the build in NetScaler release 10.1, build 118.7 or later. See Citrix NetScaler for the documentation.
7. Extract the contents of the installation package. Example:
   tar –xvzf build_10.1-49.3_nc.tgz
8. Run the installns script to install the new version of the system software. The script updates the /etc directory. Example:
   ./installns
   Note:
   To install a FIPS appliance, run the installns script with the -F option. To automatically clean up the flash, run the installns script with the -c option.

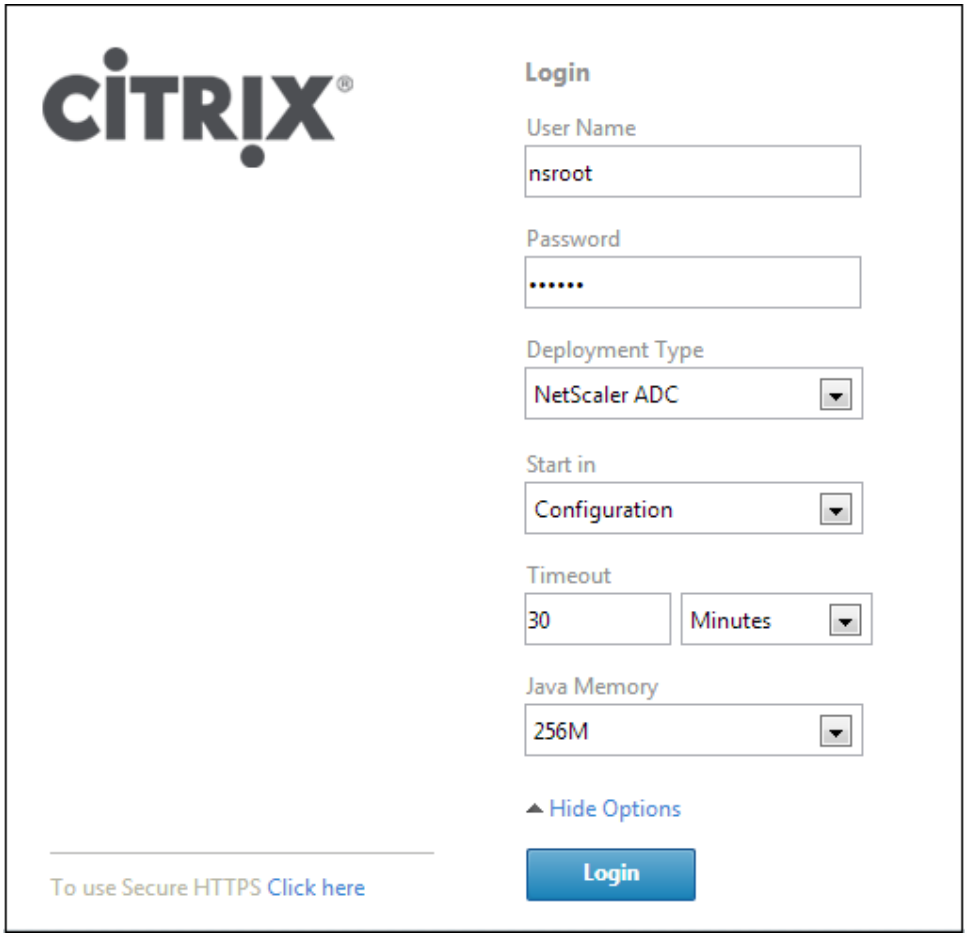
   Warning: When upgrading to the NetScaler nCore build, the installation script prompts you to delete the /var directory if the swap partition is smaller than 32 gigabytes (GB). If this prompt appears, type N, save any important files located in /var to a backup location, and then re-run the installation script.
   If the free space available on the flash drive is insufficient to install the new build, the appliance prompts you to initiate a cleanup of the flash drive. For more information, see "Auto Cleanup".
9. When prompted, restart the NetScaler.
10. (Optional) If you performed step 4, do the following:
    1. Manually compare the files in /var/nsconfig_backup and /etc and make appropriate changes in /etc.
    2. To maintain persistency, move the updated files in /etc to /nsconfig.
    3. Restart the appliance to put the changes into effect.

login: nsroot

```
Password: nsroot
Last login: Mon Apr  19 03:37:27 2008 from 10.102.29.9
Done
> save config
> shell
Last login: Mon Apr  10 03:51:42 from 10.103.25.64
root@NSnnn# cd /var/nsinstall
root@NSnnn# cd 10_1nsinstall
root@NSnnn# mkdir build_110
root@NSnnn# cd build_110
root@NSnnn# ftp ... get  build-10.1-110_nc.tgz
root@NSnnn# tar xzvf build-10.1-110_nc.tgz
root@NSnnn# ./installns
installns version (10.1-110) kernel (ns-10.1-110_nc.gz)
...
...
...
Copying ns-10.1-110_n.gz to /flash/ns-10.1-110_nc.gz ...

...
Installation has completed.

Reboot NOW? [Y/N] Y
```

To upgrade a standalone NetScaler running release 8.0, 8.1, 9.0, 9.1, 9.2, 9.3, or 10 by using the configuration utility

1. In a Web browser, type the IP address of the NetScaler, such as http://10.102.29.50.
2. In User Name and Password, type the administrator credentials.
3. In Deployment Type, select NetScaler ADC.
4. In Start in, select Configuration, and then click Login, as shown in the following figure.



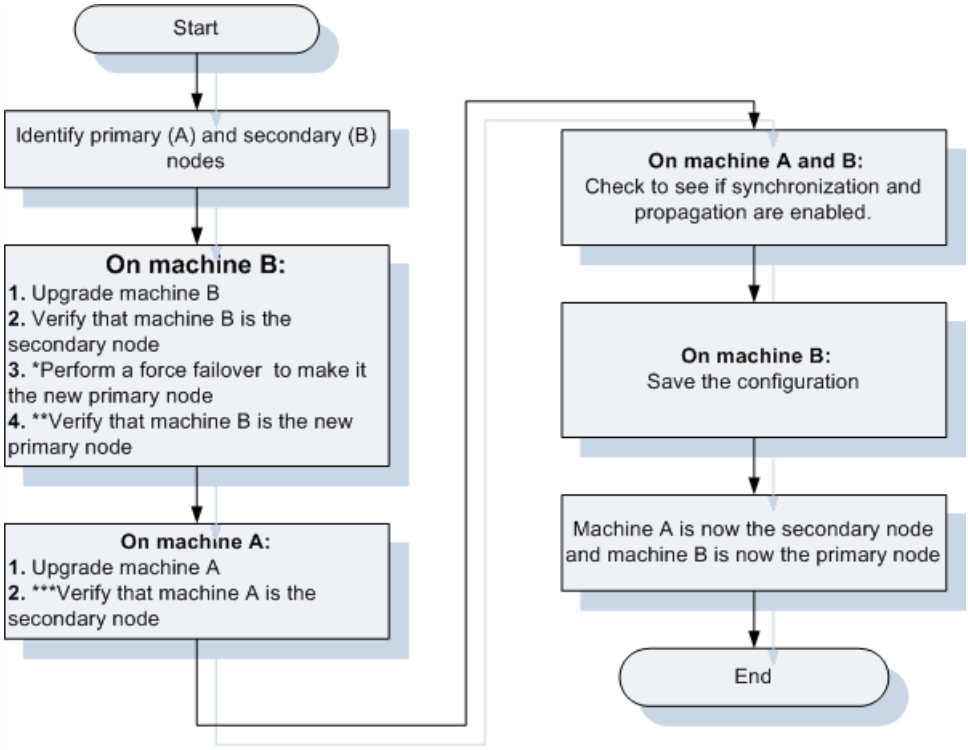
5. In the configuration utility, in the navigation pane, click System.
6. In the System Overview page, click Upgrade Wizard.
7. Follow the instructions to upgrade the software.
8. When prompted, select Reboot.
   Note: After the upgrade, close all browser instances and clear your computer's cache before accessing the appliance.

Updated: 2015-04-1

To upgrade the system software on NetScaler units in a high availability (HA) pair, first upgrade the secondary node, and then the primary node.
Note: If the two nodes in an HA configuration are running different NetScaler software releases, the following information does not get synchronized on the primary and secondary nodes:

- States of the services
- Connection failover sessions
- Persistence sessions

Figure 1. Upgrading a High Availability Pair



*After upgrading machine B, it becomes the primary node. If machine B does not function as expected, enter the force failover command on the new primary node (machine B) forcing it to again become the secondary node, and contact Citrix Customer Service before proceeding. After you test that machine B properly functions as the new primary node, proceed with upgrading the former primary node (machine A).

**After machine B is upgraded successfully, both synchronization and propagation are automatically disabled until you upgrade machine A.

***After both the nodes are upgraded successfully, synchronization and propagation are automatically enabled.

In the following procedure, machine A is the primary node and machine B is the secondary node before the upgrade.

To upgrade NetScaler units in a high availability pair running release 8.0, 8.1, 9.0, 9.1, 9.2, 9.3, or 10 by using the command line interface

On machine B (original secondary node)

1. Follow the procedure for upgrading a standalone node as described in "Upgrading a Standalone NetScaler Appliance". The procedure includes optional steps to avoid losing any updates that are pushed to the /etc directory during the upgrade.
2. After the appliance restarts, log on with the administrator credentials and enter the show ha node command to verify that the appliance is a secondary node.
3. Test the new build by entering the force failover command on the secondary node (machine B). At the command prompt type force failover.
   When you do so, machine B becomes the primary node. If machine B does not function as expected, enter the force failover command on the new primary node (machine B) forcing it to again become the secondary node, and contact Citrix Customer Service before proceeding.
4. Enter the show ha node command to verify that machine B is the new primary node.
   Example

```
login: nsroot
Password: nsroot
Last login: Mon Mar  26 08:37:26 2008 from 10.102.29.9
Done
show ha node
        2 nodes:
1)      Node ID:      0
        IP:       10.0.4.2
        Node State: UP
        Master State: Primary
        ...
        Sync State: AUTO DISABLED
        Propagation: AUTO DISABLED
        ...
Done
```

Note: After machine B is upgraded successfully, both synchronization and propagation are automatically disabled until you upgrade machine A.

On machine A (original primary node)

5. Follow the procedure for upgrading a standalone node as described in "Upgrading a Standalone NetScaler Appliance." The procedure includes optional steps to avoid losing any updates that are pushed to the /etc directory during the upgrade.
6. After the appliance restarts, log on by using the administrator credentials, and enter the show ha node command to verify that the appliance is a secondary node and that synchronization is disabled.
   Note: After both nodes are upgraded successfully, synchronization and propagation are automatically enabled.
   On machine A and machine B
7. After successfully upgrading both the nodes, run the show ha node command to verify that synchronization and propagation are enabled on the primary node and synchronization is successful and propagation is enabled on the secondary node.
   Example

   On Primary node (Machine B)

```
show ha node
        Node ID:      0
        IP:   10.0.4.2
        Node State: UP
        Master State: Primary
...
...
        INC State: DISABLED
        Sync State: ENABLED
        Propagation: ENABLED
        Enabled Interfaces : 1/1
        Disabled Interfaces : None
        HA MON ON Interfaces : 1/1
...
```

...
Local node information
Critical Interfaces: 1/1
Done

On Secondary node (Machine A)
Show ha node
Node ID: 0
IP: 10.0.4.11
Node State: UP
Master State: Secondary
..
..
INC State: DISABLED
Sync State: SUCCESS
Propagation: ENABLED
Enabled Interfaces : 1/1
Disabled Interfaces : None
HA MON ON Interfaces : 1/1
. . .
. . .
Local node information:
Critical Interfaces: 1/1
Done
On machine B (new primary node)

8. Enter the save ns config command to save the configuration.

Machine B (original secondary node) is now the primary node and machine A (original primary node) is now the secondary node.

Note: You can enter the force failover command again to make machine A (original primary node) as the primary node and machine B (original secondary node) as the secondary node.

To upgrade NetScaler units in a high availability pair running release 8.0, 8.1, 9.0, 9.1, 9.2, 9.3, or 10 by using the configuration utility

1. Log on to the secondary node and perform the upgrade as described in "To upgrade a standalone NetScaler running release 8.0, 8.1, 9.0, 9.1, 9.2, or 9.3 by using the configuration utility.
   Note: Before upgrading the primary node (machine A), you have the option to test the new release by entering the force failover command at the command line interface on the secondary node (machine B). When you do so, machine B becomes the primary node. If machine B does not function as expected, enter the force failover command at the command line interface on the new primary node (machine B) forcing it to again become the secondary node, and contact Citrix Customer Service before proceeding. If machine B properly assumes the role of primary node, proceed with upgrading the former primary node (machine A).
2. Log on to the primary node and perform the upgrade as described in "To upgrade a standalone NetScaler running release 8.0, 8.1, 9.0, 9.1, 9.2, or 9.3 by using the configuration utility".

Upgrading to a Later Build within Release 10.1

Oct 30, 2015

To upgrade from an earlier 10.1 build to a later 10.1 build on a standalone NetScaler appliance or a high availability pair, you can use the configuration utility or the command line interface. You use the same basic procedure to upgrade either a standalone appliance or each appliance in a high availability pair, although additional considerations apply to upgrading a high availability pair.

This document includes the following information:

- Upgrading a Standalone NetScaler Appliance to a Later Build
- Upgrading a NetScaler High Availability Pair to a Later Build

Upgrading a Standalone NetScaler Appliance to a Later Build

Updated: 2014-06-13

In the following procedure, <targetbuildnumber> is the build number that you are upgrading to within the 10.1 release. The procedure includes optional steps to avoid losing any updates that are pushed to the /etc directory during the upgrade.

To upgrade a standalone NetScaler appliance running release 10.1 to a later build by using the command line interface

1. Use an SSH client, such as PuTTy, to open an SSH connection to the appliance.
2. Log on to the appliance by using the administrator credentials, and save the running configuration. At the prompt, type:
   save ns config
3. Create a copy of the ns.conf file. At the shell prompt, type:
   1. cd /nsconfig
   2. cp ns.conf ns.conf.NS<releasenumber><currentbuildnumber>

   You should backup the configuration file to another computer.
4. (Optional) If you have modified any of the following files in the /etc directory, and copied them to /nsconfig to maintain persistency, any updates that are pushed to the /etc directory during the upgrade might be lost:
   - ttys
   - resolv.conf
   - sshd_config
   - host.conf
   - newsyslog.conf
   - host.conf
   - httpd.conf
   - rc.conf
   - syslog.conf
   - crontab
   - monitrc

   To avoid losing these updates, create a /var/nsconfig_backup directory, and move the customized files to this directory. That is, move any files that you modified in /etc directory and copied to /nsconfig, by running the following command:

   cp /nsconfig/<filename> /var/nsconfig_backup

   Example:

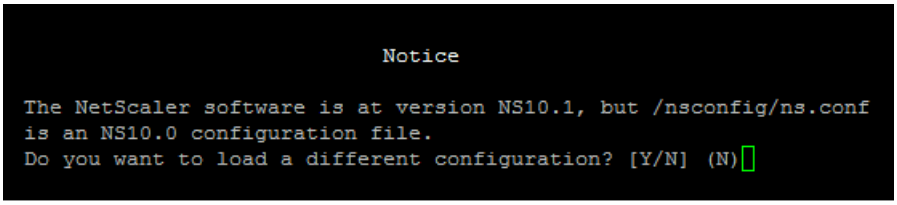
cp /nsconfig/syslog.conf /var/nsconfig_backup

5. Create a location for the installation package. At the shell prompt, type:
   1. cd/var/nsinstall
   2. mkdir <releasenumber>nsinstall
   3. cd <releasenumber>nsinstall
   4. mkdir build_<targetbuildnumber>
   5. cd build_<targetbuildnumber>
6. Download or copy the installation package (build-10.1-<targetbuildnumber>_nc.tgz) to the directory that you created for it. To download the installation package from the Citrix Web site, do the following:
   1. Go to MyCitrix.com, log on with your credentials, and click Downloads.
   2. In the Select a Product, select NetScaler ADC.
   3. Under Firmware, click the release and build number to download.
   4. Click Get Firmware.

   Note: Documentation is not available as part of build in NetScaler release 10.1, build 118.7, or later. See Citrix NetScaler for the documentation.
7. Extract the contents of the installation package. Example:
   tar –xvzf build_10.1-121.10_nc.tgz
8. Run the installns script to install the new version of the system software. The script updates the /etc directory.
   Note:
   To install a FIPS appliance, run the installns script with the -F option. To automatically clean up the flash, run the installns script with the -c option.

   During the upgrade, you are prompted for an option to load a different configuration, as shown in the following figure.

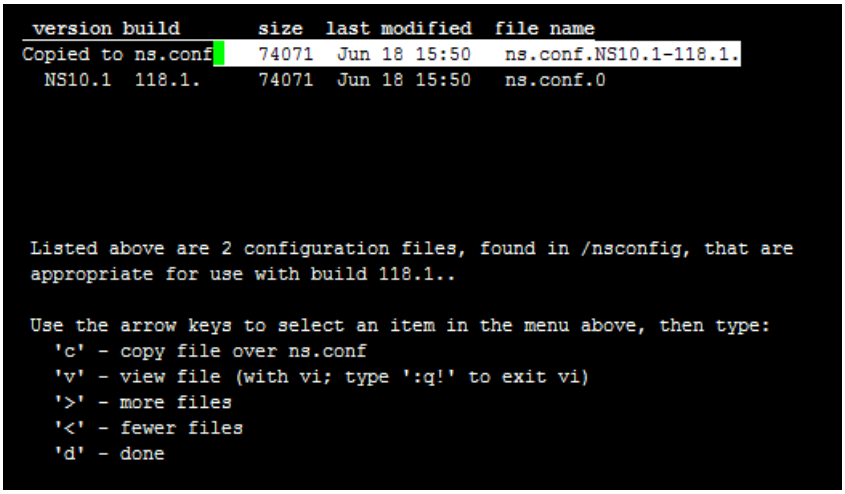
   Figure 1. Upgrade menu to load a different configuration



   If you do not want to load a different configuration and continue with the upgrade, typ N. wish to load a different configuration file, then type Y. If the configuration file for the build that you are upgrading to exists on the appliance, you are prompted to load that configuration, as shown in the following figure.

   If you want to load a different configuration file, type Y. If the configuration file for the build to which you are upgrading exists in the appliance, you are prompted to load that configuration, as shown in the following figure.

   Figure 2. Upgrade menu if configuration file exists



9. When prompted, restart the appliance.
10. (Optional) If you performed step 4, do the following:
    1. Manually compare the files in /var/nsconfig_backup and /etc and make appropriate changes in /etc.
    2. To maintain persistency, move the updated files in /etc to /nsconfig.
    3. Restart the appliance to put the changes into effect.

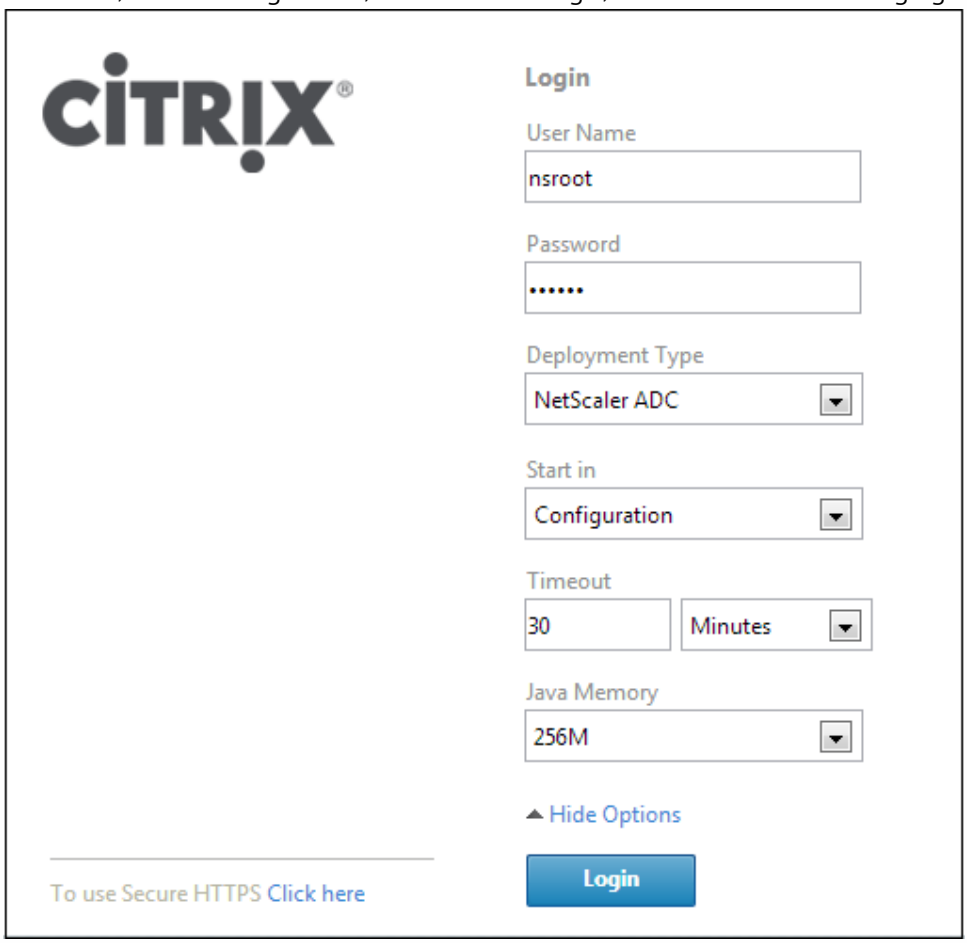
Example

```
login: nsroot
Password:
Last login: Thu Aug  9 12:12:54 2012 from 10.144.7.22
Done
> save ns config
> shell
Last login: Mon Aug  9 03:51:42 from 10.103.25.64
root@NSnnn# cd /var/nsinstall
root@NSnnn# cd 10.1nsinstall
root@NSnnn# mkdir build_118.7
root@NSnnn# cd build_118.7
root@NSnnn# ftp ... get build-10.1-118.7_nc.tgz
root@NSnnn# tar  build-10.1-118.7_nc.tgz
root@NSnnn# ./installns
installns version (10.1-118.7) kernel (ns-10.1-118.7_nc.gz)
The Netscaler version 10.1-118.7 checksum file is located on
http://www.mycitrix.com under Support > Downloads > Citrix NetScaler.
Select the Release 10.1-118.7 link to view the MD5 checksum file for build 10.1-118.7.

There may be a pause of up to 3 minutes while data is written to the flash.
Do not interrupt the installation process once it has begun....
...
...
Copying ns-10.1-118.7_nc.gz to /flash/ns-10.1-118.7_nc.gz ...
...
Installation has completed.
```

Reboot NOW? [Y/N] Y

To upgrade a standalone NetScaler running release 10.1 to a later build by using the configuration utility

1. In a web browser, type the IP address of the NetScaler, such as http://10.102.29.50.
2. In User Name and Password, type the administrator credentials.
3. In Deployment Type, select NetScaler ADC.
4. In Start in, select Configuration, and then click Login, as shown in the following figure.

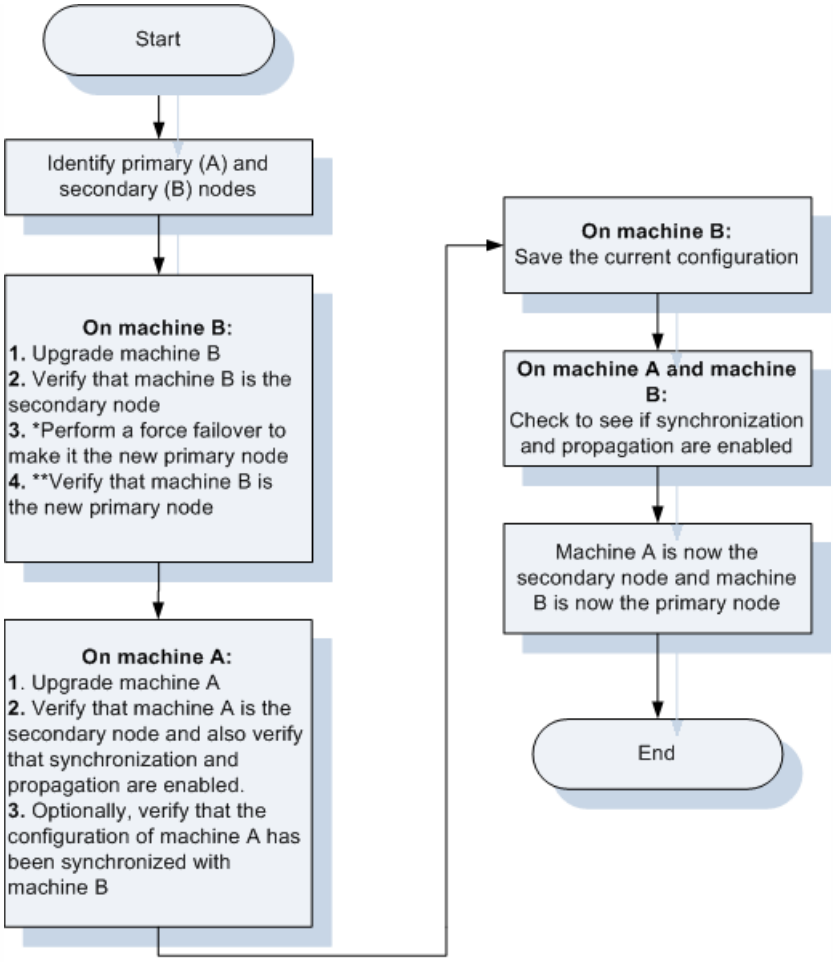


5. In the configuration utility, in the navigation pane, click System.
6. In the System Overview page, click Upgrade Wizard.
7. Follow the instructions to upgrade the software.
8. When prompted, select Reboot.
   Note: After the upgrade, close all browser instances and clear your computer's cache before accessing the appliance.

Upgrading a NetScaler High Availability Pair to a Later Build

Updated: 2014-12-10

To upgrade the system software on NetScaler appliances in a high availability (HA) pair, upgrade the secondary node first, and then upgrade the primary node.
Warning: In certain cases, after you upgrade one of the nodes in an HA pair, synchronization and propagation are automatically disabled until you upgrade the other node. To determine whether synchronization and propagation are disabled, at the command line interface, type: show ha node
Figure 3. Upgrading a NetScaler High Availability Pair to a Later Build



*After upgrading machine B, it becomes the primary node. If machine B does not function as expected, enter the force failover command on the new primary node (machine B) forcing it to again become the secondary node, and contact Citrix Customer Service before proceeding. After you test that machine B properly functions as the new primary node, proceed with upgrading the former primary node (machine A).

**After machine B is upgraded successfully, both synchronization and propagation are automatically disabled until you upgrade machine A.

In the following procedure, machine A is the original primary and machine B is the original secondary node, and <targetbuildnumber> is the build number that you are upgrading to within the 10.1 release.

To upgrade a NetScaler high availability pair to a later build by using the command line interface

On machine B (original secondary node)

1. Follow the procedure for upgrading a standalone node as described in "Upgrading a Standalone NetScaler Appliance to a Later Build". The procedure includes optional steps to avoid losing any updates that are pushed to the /etc directory during the upgrade.
2. After the NetScaler restarts, log on by using the administrator credentials and enter the show ha node command to verify that the appliance is a secondary node.
3. Test the new build by entering the force failover command on the secondary node (machine B). At the command prompt type force failover.
   When you do so, machine B becomes the primary node. If machine B does not function as expected, enter the force failover command on the new primary node (machine B) forcing it to again become the secondary node, and contact

Citrix Customer Service before proceeding.

4. Enter the show ha node command to verify that machine B is the new primary node.
   On machine A (original primary node)

5. Follow the procedure for upgrading a standalone node as described in "Upgrading a Standalone NetScaler Appliance to a Later Build." The procedure includes optional steps to avoid losing any updates that are pushed to the /etc directory during the upgrade.
6. After the appliance restarts, log on by using the administrator credentials and enter the show ha node command to verify that the appliance is a secondary node and that synchronization and propagation are enabled. Optionally, enter the show ns runningconfig command on both the nodes and compare the result to verify that the configuration of machine A has been synchronized with that of machine B.
   On machine B (new primary node)

7. Enter the save ns config command to save the current configuration.
   On machine A and machine B

8. After successfully upgrading both the nodes, run the show ha node command to verify that synchronization and propagation are enabled.
   Example

```
show ha node
        Node ID:      0
        IP:   10.0.4.2
        Node State: UP
        Master State: Primary
...
...
        INC State: DISABLED
        Sync State: ENABLED
        Propagation: ENABLED
        Enabled Interfaces : 1/1
        Disabled Interfaces : None
        HA MON ON Interfaces : 1/1
...
...
        Local node information
        Critical Interfaces: 1/1
Done


Show ha node
        Node ID:      0
        IP:   10.0.4.11
        Node State: UP
        Master State: Secondary
        ..
       ..
```

```
        INC State: DISABLED
        Sync State: SUCCESS
        Propagation: ENABLED
        Enabled Interfaces : 1/1
        Disabled Interfaces : None
        HA MON ON Interfaces : 1/1
        . . .
        . . .
        Local node information:
        Critical Interfaces: 1/1
Done
```

Machine B (original secondary node) is now the primary node and machine A (original primary node) is now the secondary node.

Downgrading from Release 10.1

Jun 09, 2014

You can downgrade to any release on a standalone NetScaler or a high availability pair by using the command line interface.

Caution: Loss in configuration may occur when downgrading. You should compare the configurations before and after the downgrade, and then manually reenter any missing entries.

This procedure provides steps to downgrade from release 10.1 to an earlier release. For downgrading to an earlier build within release 10.1, see "Downgrading to an Earlier Build within Release 10.1".

Note: Downgrading using the configuration utility is not supported.

This document includes the following information:

- Downgrading a Standalone NetScaler
- Downgrading a High Availability Pair

Downgrading a Standalone NetScaler

Updated: 2015-01-09

In the following procedure, <release> and <releasenumber> represent the release version you are downgrading to, and <targetbuildnumber> represents the build number that you are downgrading to. Refer to the table below for specific values.

Table 1. Release Version Values

Release Version	<release>	<releasenumber>
10	10	10
9.3	9.3	9.3
9.2	9.2	9.2
9.1	9.1	9.1
8.1	rhodes	8.1
8.0	andes	8.0
7.0	sierra	7.0

To downgrade a standalone NetScaler

1. Open an SSH connection to the NetScaler by using an SSH client, such as PuTTY.
2. Log on to the NetScaler by using the administrator credentials. Save the running configuration. At the prompt, type:

save config

3. Create a copy of the ns.conf file. At the shell prompt, type:
   1. cd /nsconfig
   2. cp ns.conf ns.conf.NS10.1<currentbuildnumber>

   You should backup a copy of the configuration file on another computer.
4. Copy the <releasenumber> configuration file (ns.conf.NS<releasenumber>) to ns.conf. At the shell prompt, type:

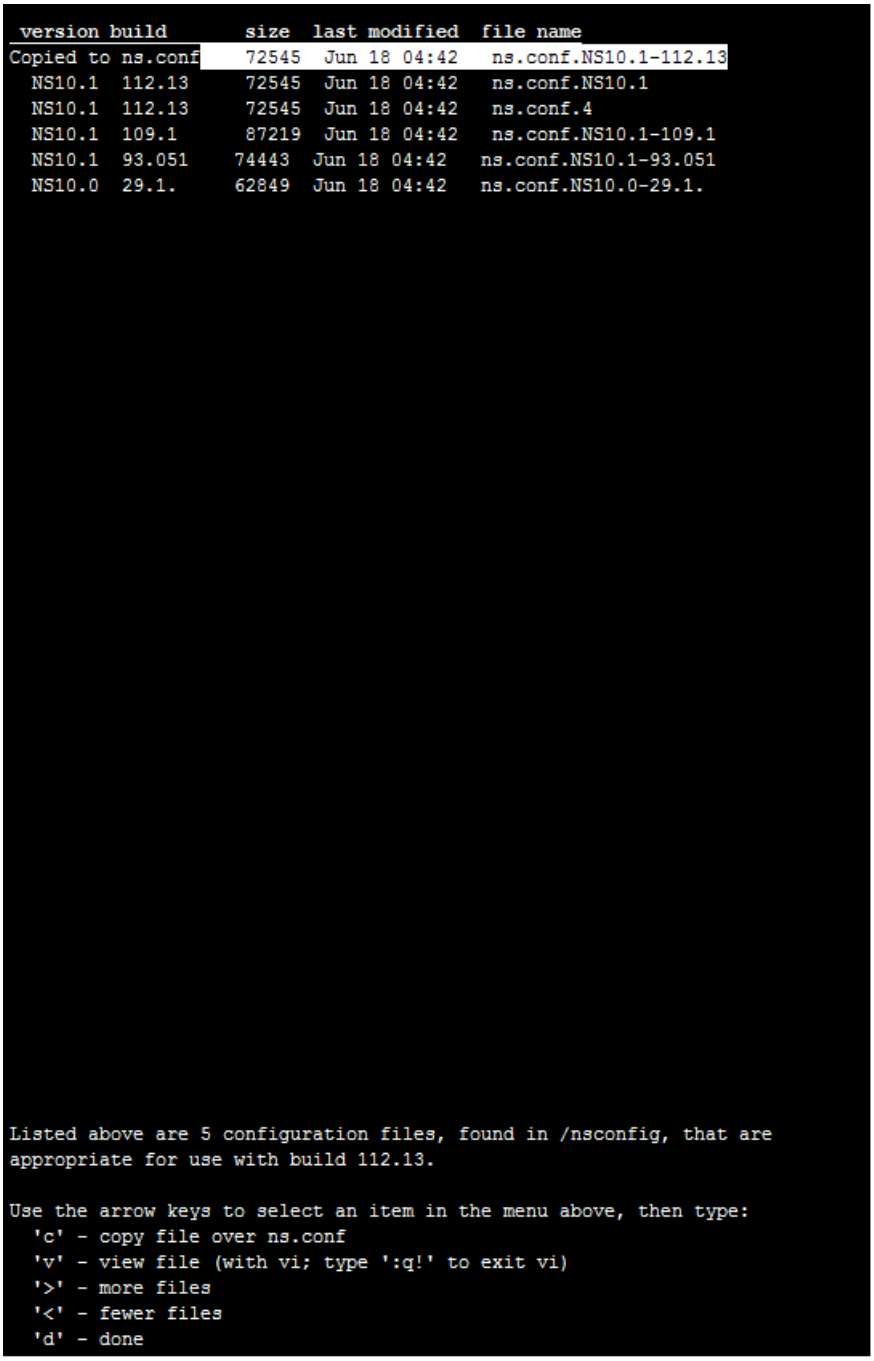
   cp **ns.conf.NS**<releasenumber> **ns.conf**

   Note: ns.conf.NS<releasenumber> is the backup configuration file that is automatically created when the system software is upgraded from release version <releasenumber> to the current release version. There may be some loss in configuration when downgrading. After the appliance restarts, compare the configuration saved in step 3 with the running configuration, and make any adjustments for features and entities configured before the downgrade. Save the running configuration after making the changes.

   Important: If routing is enabled, perform step 5. Otherwise, skip to step 6.
5. If routing is enabled, the ZebOS.conf file will contain the configuration. At the shell prompt, type:
   1. cd **/nsconfig**
   2. cp **ZebOS.conf ZebOS.conf.NS**10.1
   3. cp ZebOS.conf.NS<targetreleasenumber> ZebOS.conf
6. Change directory to /var/nsinstall/<releasenumber>nsinstall, or create one if it does not exist.
7. Change directory to build_<targetbuildnumber>, or create one if it does not exist.
8. Download or copy the installation package (build-<release>-<targetbuildnumber>.tgz) and the documentation bundle (ns-<releasenumber>-<targetbuildnumber>-doc.tgz) to this directory and extract the contents of the installation package.
9. Run the installns script to install the new version of the system software. The script updates the /etc directory.

   If the configuration file for the build that you are downgrading to exists on the appliance, you are prompted to load that configuration, as shown in the following figure.

   Figure 1. Downgrade menu if configuration file exists



If the free space available on the flash drive is insufficient to install the new build, the NetScaler prompts you to initiate a cleanup of the flash drive. For more information, see "Auto Cleanup".

10. When prompted, restart the NetScaler.

Example

```
login: nsroot
Password: nsroot
Last login: Tue Mar  27 01:38:25 2008 from 10.102.29.9
Done
> save config
> shell
Last login: Tue Mar  27 02:07:06 from 10.103.25.64
```



```
root@NSnnn# cp ns.conf.NS10 ns.conf
root@NSnnn# cd /var/nsinstall
root@NSnnn# mkdir 10nsinstall
root@NSnnn# cd 10nsinstall
root@NSnnn# mkdir build_55
root@NSnnn# cd build_55
root@NSnnn# ftp ... get build-10.0-55_nc.tgz
root@NSnnn# get ns-10-55-doc.tgz
root@NSnnn# tar xzvf build-10.0-55_nc.tgz
root@NSnnn# ./installns
installns version (10.0-55) kernel (ns-10.0-55.gz)
...
...
...
Copying ns-10.0-55.gz to /flash/ns-10.0-55_nc.gz ...
Changing /flash/boot/loader.conf for ns-10.0-55 ...
Installing documentation...

Installation has completed.

Reboot NOW? [Y/N] Y
```

Downgrading a High Availability Pair

Updated: 2013-08-21

To downgrade the system software on NetScaler units in a high availability pair, you need to downgrade the software first on the secondary node and then on the primary node. For instructions on downgrading each node separately, see "Downgrading a Standalone NetScaler".

Downgrading to an Earlier Build within Release 10.1

Jan 29, 2014

You can downgrade from a later 10.1 build to an earlier 10.1 build on a standalone NetScaler or a high availability pair. This procedure must be performed by using the command line interface.

Warning: Loss in configuration may occur when downgrading. You should compare the configurations before and after the downgrade, and then manually readd any missing entries.

This document includes the following information:

- Downgrading a Standalone NetScaler to an Earlier Build
- Downgrading a NetScaler High Availability Pair to an Earlier Build

Downgrading a Standalone NetScaler to an Earlier Build

Updated: 2014-01-29

In the procedure below, <targetbuildnumber> is the build number that you are downgrading to within the same release.

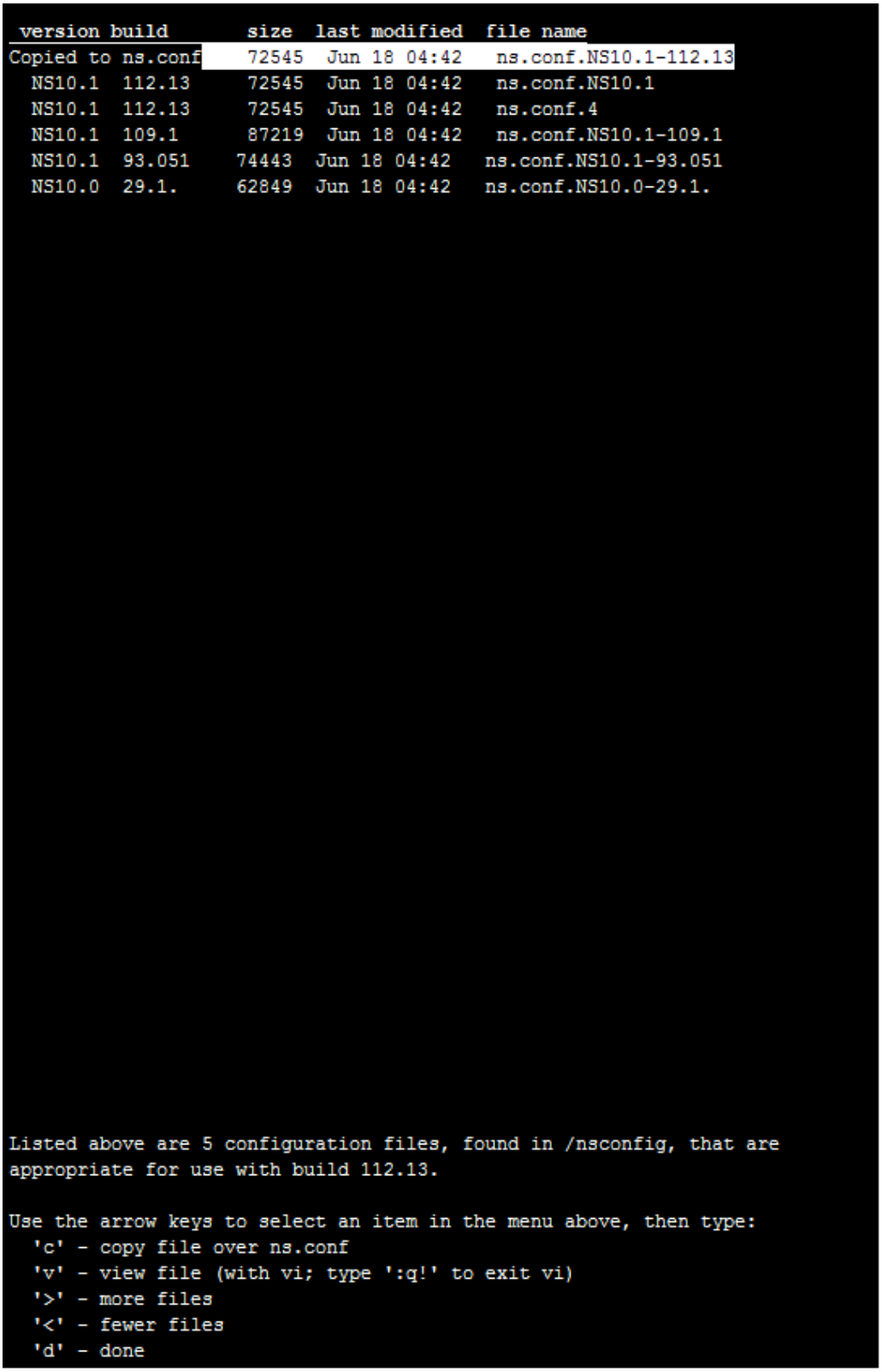
To downgrade a standalone NetScaler to an earlier build

1. Use an SSH client, such as PuTTy, to open an SSH connection to the appliance.
2. Log on to the NetScaler by using the administrator credentials. Save the running configuration. At the prompt, type:

   save ns config

   Caution: If ns.conf.NS10.1-<targetbuildnumber> does not exist, loss in configuration may occur when downgrading to an earlier build. The errors and warnings appear only on the console. Please watch the console closely for these errors and warnings. After the appliance restarts, compare the saved configuration with the running configuration, and make any adjustments for features and entities configured before the downgrade. Save the running configuration after making the changes.
3. Change directory to /var/nsinstall/10.1nsinstall.
4. Change directory to build_<targetbuildnumber>, or create one if it does not exist.
5. Download or copy the installation package (build-10.1-<targetbuildnumber>_nc.tgz) and the documentation bundle (ns-9.2-<targetbuildnumber>.e-doc.tgz) to this directory and extract the contents of the installation package.
   Note: Documentation is not included in this NetScaler build. It is available, only till the NetScaler release 10.1, build 113.x and later.
6. Run the installns script to install the old version of the system software. The script updates the /etc directory.
   If the configuration file for the build that you are downgrading to exists on the appliance, you are prompted to load that configuration, as shown in the following figure.

   Figure 1. Downgrade menu if configuration file exists



7. When prompted, restart the NetScaler.

Example

login: nsroot
Password: nsroot

```
Last login: Sun May  5 08:38:25 2013 from 10.102.29.4
Done
> save ns config
> shell
Last login: Sun Aug  5 09:07:06 from 10.103.25.64
root@NSnnn# cp ns.conf.NS10.1-112.13 ns.conf
root@NSnnn# cd /var/nsinstall
root@NSnnn# cd 10.1nsinstall
root@NSnnn# cd build_112_13
root@NSnnn# ftp ... get build-10.1-112.13_nc.tgz
root@NSnnn# tar xzvf build-10.1-112.13_nc.tgz
root@NSnnn# ./installns
installns version (10.1-112.13) kernel (ns-10.1-112.13.gz)
...
...
...
Copying ns-10.1-112.13_nc.gz to /flash/ns-10.1-112.13_nc.gz ...
Changing /flash/boot/loader.conf for ns-10.1-112.13 ...

Installation has completed.

Reboot NOW? [Y/N] Y
```

Downgrading a NetScaler High Availability Pair to an Earlier Build

Updated: 2014-01-29

To downgrade the system software on NetScaler units in a high availability pair, you need to downgrade the software first on the secondary node and then on the primary node. For instructions on downgrading each node separately, see "Downgrading a Standalone NetScaler to an Earlier Build".

Note: In an HA setup, both nodes must run NetScaler nCore or NetScaler classic. If the nodes are running NetScaler classic and you want to migrate to NetScaler nCore of the same NetScaler release, propagation and synchronization are not supported during the migration process. Once migration is complete, you have to manually enable propagation and synchronization. The same applies if you migrate from NetScaler nCore to NetScaler classic.

Auto Cleanup

Jul 01, 2014

The cleanup procedure has been simplified in the later versions of release 7.0 (build 48 and later) and in releases 8.0, 8.1, 9.0, 9.1, 9.2, 9.3, 10, and 10.1. You no longer have to manually delete build files from the flash drive. During the installation process, if the free space on the flash drive is found to be insufficient, the NetScaler prompts you to initiate the cleanup process.

Note: To automatically clean up the flash, run the installns script with the -c option.
When downgrading to release 7.0, the prompt looks like this:

```
Installation path for kernel will be /flash
Size of kernel ns-7.0-21.7.gz is 58003.323 kilobytes
Available space on /flash/ filesystem is 25075 kilobytes
Available space on /flash/ filesystem is insufficient to install ns-7.0-21.7.gz
Do you want Auto Cleanup [Y/N] ?
```

When upgrading to release 8.1, the prompt looks like this:

```
Installation path for kernel is /flash
Size of kernel ns-8.1-32.2.gz is 61062.235 kilobytes
Available space on /flash/ filesystem is 59108 kilobytes
Available space on /flash/ filesystem is insufficient to install ns-8.1-32.2.gz
Do you want installns to free space by archiving older releases? [Y/ N]
```

To initiate the cleanup process, press Y. Messages similar to the following appear:

```
Archiving older releases ...
  Creating the archive directory /var/nsbackup/ns_2007_2_16_1_6_26 ...
  Move //flash//ns-6.1-97.4.m.gz /var/nsbackup/ ns_2007_2_16_1_6_26ns-6.1-97.4.m.gz ...
  Move //flash//ns-8.1-32.2.gz /var/nsbackup/ns_2007_2_16_1_6_26ns-8.1-32.2.gz ...
Archive operation completed, free space is 156452, required space is 61062.235
```

The installation process automatically continues after successful completion of the cleanup.

Troubleshooting

Aug 13, 2013

If the appliance does not work as expected after you complete the installation, upgrade, or downgrade process, the first thing to do is to check for the most common causes of the problem.

This document includes the following information:

- Resources for Troubleshooting
- Troubleshooting Issues Related to the Installation, Upgrade, and Downgrade processes

Resources for Troubleshooting

Updated: 2013-08-13

For best results, use the following resources to troubleshoot an issue related to installing, upgrading, or downgrading a NetScaler appliance:

- The configuration files from the appliance. In case of a High Availability pair, the configuration files from both appliances.
- The following files from the appliance(s):
  - The relevant newnslog files.
  - The ns.log file.
  - The messages file.
- A network topology diagram.

Troubleshooting Issues Related to the Installation, Upgrade, and Downgrade processes

Updated: 2013-11-22

Following are the most common installation, upgrade, and downgrade issues, and tips for resolving them:

- **Issue**
  The NetScaler appliance is not accessible after the software downgrade

  Cause

  During the software downgrade process, if the configuration file of the existing release and build does not match the configuration file of the earlier release and build, the appliance cannot load the configuration, and the default IP address is assigned to the appliance.

  Resolution

  - Verify that the appliance is accessible from the console.
  - Verify the NSIP address and the routes on the appliance.
    - If the IP address has changed to the default 192.168.100.1 IP address, change the IP address as required.
    - Verify that the appliance is accessible.
- **Issue**
  Configuration of the appliance is lost after you upgrade the software across multiple releases.

  Cause

  Some migration commands are built in for upgrading to the next release. Such commands might not be available across releases.

Resolution

- Verify the path of the upgrade process. Citrix recommends upgrading by one release at a time. For example, if the softer needs to be upgraded from NetScaler release 9.2 to NetScaler release 10.1, the following is the recommended path for the upgrade:
  - NetScaler release 9.2 to NetScaler release 9.3
  - NetScaler release 9.3 to NetScaler release 10
  - NetScaler release 10 to NetScaler release 10.1
- Verify that the appliance has appropriate license files.
- Verifying the configuration at each step of the upgrade process can give you pointers to the issue.

- **Issue**
  During an upgrade, if I run the command for synchronizing, the following message appears:

  Command failed on the secondary node but succeeded on the primary node.

  Resolution

  Do not run any dependent commands (set /unset /bind /unbind) when High Availability (HA) synchronization is in progress.

- **Issue**
  During an upgrade process, traffic does not pass through the new primary node when you run the force failover command.

  Resolution
  - Check for problems with the network topology and the switch configurations.
  - Run the set L2param -garpreply ENABLED command to enable the GARP reply.
  - Try using VMAC if not already used.
  - Run the sendarp –a command from the primary node.
- **Issue**
  In an HA pair, after you run the force HA failover command the devices keep rebooting. The secondary device does not come up after an upgrade.

  Resolution

  Check to see if the /var directory is full. If so, remove the old installation files. Run the df –h command to show the available disk space.

- **Issue**
  After upgrading an HA pair, one of the nodes is listed as state UNKNOWN.

  Resolution
  - Check to see if both nodes are running the same build. If the builds are not same and HA nodes have a version mismatch, some of the fields are shown as UNKNOWN when you run the show ha node command.
  - Check to see if the secondary appliance is reachable.
- **Issue**
  After you upgrade the NetScaler appliance, the interface shows most of the load balancing virtual servers and services are DOWN.

  Resolution

Verify that the SNIP address is active on the secondary appliance. Also, type the show service command to see if the service is running.

- **Issue**
  After performing an upgrade, all virtual servers are down on the secondary appliance.

  Resolution

  Enable the HA state and HA synchronization by running the following commands:
  - set node hastate enable
  - set node hasync enable

  Disabling HA is not recommended.
- **Issue**
  After performing a downgrade, the NetScaler appliance does not boot up properly.

  Resolution

  Check to see if the correct license has been installed.

- **Issue**
  In an HA pair, some features do not get synchronized after an upgrade is performed.

  Resolution

  Run the sync ha file misc command to synchronize the configurations files from the primary node to secondary node.

- **Issue**
  During reboot, the following error message appears:

  `One or more commands in ns.conf failed`What should I do?

  Resolution

  Make sure that no command in the ns.conf file exceeds the 255 byte limit. In commands that create policies that are too long for the 255-byte limit, you can use pattern sets to shorten the policies.

  Example:
  `add cs policy p11 -rule 'HTTP.REQ.URL.ENDSWITH_ANY("ctx_file_extensions")'`
  `Done`
  ctx_file_extensions is a default patset that covers a large number of extensions. In addition to the default pattern sets, you can create user-defined pattern sets. Add a patset by running the following command:
  `add patset <name>`
  Note: Patsets are supported only in release 9.3 or later.
- **Issue**
  When upgrading a NetScaler VPX appliance, I am told to free up space in /var. What files do I remove?

  Resolution

  Remove the old installation files from /var/tmp/ directory. Also remove unwanted files from /flash.

- **Issue**
  There is no connectivity to the graphical user interface (GUI) when you run the force HA failover command on the

secondary appliance.

Resolution

Log on to the secondary appliance using the command line interface and enable the access to GUI by running the set ns ip <IP> -gui enabled command.

- **Issue**
  After performing an upgrade, and when I click on any link on the GUI that has to load a java applet (Upgrade Wizard or license Wizard), the following error message appears: **GUI version does not match with the kernel version. Please close this instance, clear java plug-in cache and reopen**.

  Resolution

  - Log on to the NetScaler appliance using the GUI.
  - Navigate to NetScaler Gateway > Global Settings.
  - Click Change Global Settings under Settings.
  - In the details pane, under Client Experience, select Default from the UI theme list.
  - Click OK.

Note: These troubleshooting steps also apply to issues with configuration loss when downgrading the software across multiple releases.
For any other issue, see the release notes, Knowledge Center articles, and FAQs.

System

Sep 05, 2013

The following topics provide information of the NetScaler system.

Administration	Manages and monitors the NetScaler appliance by using built-in features such as authentication and authorization, SNMP management, audit logging, web server logging, NTP management, and Reporting tool.
AppFlow	Provides information about the reporting capabilities of AppFlow feature of the NetScaler.
AutoScale	Describes how users of Citrix CloudPlatform can use the AutoScale feature on the NetScaler appliance to enable automatic scaling of their applications.
CloudBridge Connector	Provides help in reducing the cost of moving your applications to the cloud, reduce the risk of application failure, and increase network efficiency in your cloud environment.
Clustering	A setup of multiple nCore NetScaler appliances that ensure high availability, high throughput, and scalability of a deployment of NetScaler appliances.
EdgeSight Monitoring for NetScaler	Monitors the end-user experience with web applications that are served in a NetScaler environment.
Flex Tenancy	A methodology that allows you to tune a group of NetScaler virtual appliance instances to the unique characteristics and needs of individual applications in a complex Web 2.0 setup.
High Availability	A setup of two NetScaler appliances that ensure the high availability of NetScaler appliances.
Web Interface	Provides access to Citrix XenApp and Citrix XenDesktop applications.

Basic Operations

Jun 02, 2015

Any changes you make to the configuration of a NetScaler appliance are not saved automatically. You have to save the settings manually. When an appliance is restarted, it loads the latest saved configuration.

This document includes the following details:

- Viewing and Saving Configurations
- Clearing the NetScaler Configuration

Viewing and Saving Configurations

Configurations are stored in the /nsconfig/ns.conf directory. For configurations to be available across sessions, you must save the configuration after every configuration change.

To view the running configuration by using the command line interface

At the command prompt, type:

show ns runningConfig

To view the running configuration by using the configuration utility

- Navigate to System > Diagnostics and, in the View Configuration group, click Running Configuration.

To find the difference between two configuration files by using the command line interface

At the command prompt, type:

diff ns config <configfile1> <configfile2>

To find the difference between two configuration files by using the configuration utility

- Navigate to System > Diagnostics and, in the View Configuration group, click Configuration difference.

To save configurations by using the command line interface

At the command prompt, type:

save ns config

To save configurations by using the configuration utility

On the Configuration tab, in the top-right corner, click the Save icon.

To view the saved configurations by using the command line interface

At the command prompt, type:

show ns ns.conf

To view the saved configurations by using the configuration utility

Navigate to System > Diagnostics and, in the View Configuration group, click Saved Configuration.

Clearing the NetScaler Configuration

You have the following three options for clearing the NetScaler configuration.

Basic level. Clearing your configuration at the basic level clears all settings except the following:

- NSIP, MIP(s), and SNIP(s)
- Network settings (Default Gateway, VLAN, RHI, NTP, and DNS settings)
- HA node definitions
- Feature and mode settings
- Default administrator password (nsroot)

Extended level. Clearing your configuration at the extended level clears all settings except the following:

- NSIP, MIP(s), and SNIP(s)
- Network settings (Default Gateway, VLAN, RHI, NTP, and DNS settings)
- HA node definitions

Feature and mode settings revert to their default values.

Full level. Clearing your configuration at the full level returns all settings to their factory default values. However, the NSIP and default gateway are not changed, because changing them could cause the appliance to lose network connectivity.

To clear the configuration by using the command line interface

At the command prompt, type:

clear ns config -force <level>

Example: To forcefully clear the basic configurations on an appliance.

```
clear ns config -force basic
```

To clear the configuration by using the configuration utility

- Navigate to System > Diagnostics and, in the Maintenance group, click Clear Configuration and select the configuration level to be cleared from the appliance.

Configuring Clock Synchronization

Jun 02, 2015

You can configure your NetScaler appliance to synchronize its local clock with a Network Time Protocol (NTP) server. This ensures that its clock has the same date and time settings as the other servers on your network.

You can configure clock synchronization on your appliance by adding NTP server entries to the ntp.conf file from either the configuration utility or the command line interface, or by manually modifying the ntp.conf file and then starting the NTP daemon (NTPD). The clock synchronization configuration does not change if the appliance is restarted, upgraded, or downgraded. However, the configuration does not get propagated to the secondary NetScaler in a high availability setup.

Note: If you do not have a local NTP server, you can find a list of public, open access, NTP servers at the official NTP site, http://www.ntp.org, under Public Time Servers List. Before configuring your NetScaler to use a public NTP server, be sure to read the Rules of Engagement page (link included on all Public Time Servers pages).
This document includes the following details:

- Setting Up Clock Synchronization
- Starting the NTP Daemon
- Configuring Clock Synchronization Manually

Setting Up Clock Synchronization

Updated: 2014-08-18

To configure clock synchronization, you must add NTP servers and then enable NTP synchronization.

To add an NTP server by using the command line interface

At the command prompt, type the following commands to add an NTP server and verify the configuration:

- add ntp server (<serverIP> | <serverName>) [-minpoll <positive_integer>] [-maxpoll <positive_integer>]
- show ntp server

Example

```
> add ntp server 10.102.29.30 -minpoll 6 -maxpoll 11
```

To configure an NTP server by using the configuration utility

Navigate to System > NTP Servers, and create the NTP server.

Starting the NTP Daemon

Updated: 2014-08-08

When you enable NTP synchronization, the NetScaler starts the NTP daemon and uses the NTP server entries in the ntp.conf file to synchronize its local time setting. If you do not want to synchronize the appliance time with the other servers in the network, you can disable NTP synchronization, which stops the NTP daemon (NTPD).

To enable NTP synchronization by using the command line interface

At the command prompt, type one of the following commands:

enable ntp sync

To enable NTP synchronization by using the configuration utility

Navigate to System > NTP Servers, click Action and select NTP Synchronization.

Configuring Clock Synchronization Manually

Updated: 2013-08-23

You can configure clock synchronization manually by logging on to the NetScaler and editing the ntp.conf file.

To enable clock synchronization on your NetScaler by modifying the ntp.conf file

1. Log on to the command line interface.
2. Switch to the shell prompt.
3. Copy the /etc/ntp.conf file to /nsconfig/ntp.conf, unless the /nsconfig directory already contains an ntp.conf file.
4. Check the /nsconfig/ntp.conf file for the following entries and, if they are present, remove them:
   restrict localhost

   restrict 127.0.0.2

5. Add the IP address for the desired NTP server to the /nsconfig/ntp.conf file, beneath the file's server and restrict entries.
   Note: For security reasons, there should be a corresponding restrict entry for each server entry.
6. If the /nsconfig directory does not contain a file named rc.netscaler, create the file.
7. Add the following entry to /nsconfig/rc.netscaler: /usr/sbin/ntpd -c /nsconfig/ntp.conf -l /var/log/ntpd.log &
   This entry starts the ntpd service, checks the ntp.conf file, and logs messages in the /var/log directory.

   This process runs every time the NetScaler is restarted.

8. Reboot the NetScaler to enable clock synchronization.
   Note:
   If you want to start the time synchronization process without restarting the NetScaler, run the following command from the shell prompt:

   /usr/sbin/ntpd -c /nsconfig/ntp.conf -l /var/log/ntpd.log &

Viewing the System Date and Time

Aug 08, 2014

To change the system date and time, you must use the shell interface to the underlying FreeBSD OS. However, to view the system date and time, you can use the command line interface or the configuration utility.

To view the system date and time by using the command line interface

At the command prompt, type:

```
show ns config
```

To view the system date and time by using the configuration utility

Navigate to System and select the System Information tab to view the system date.

Backing up and Restoring the NetScaler Appliance

Sep 15, 2015

You can back up the current state of a NetScaler appliance, and later use the backed up files to restore the appliance to the same state. You must use this feature before performing an upgrade or for precautionary reasons. A backup of a stable system enables you to restore the system to a stable point in the event that it becomes unstable.

Points to remember

- You cannot use the backup file taken from one appliance to restore a different appliance.
- You can back up and restore appliances in an HA setup, but make sure that you restore to the same appliance from which the backup file was created. For example, if the backup was taken from the primary appliance of the HA pair, when restoring make sure that the appliance you are restoring is the same appliance, even if it is no longer the primary appliance.
- You cannot perform the backup and restore operation on a NetScaler cluster.

This document includes the following details:

Backing up a NetScaler Appliance

Updated: 2015-03-11

Depending on the type of data to be backed up and the frequency at which you will create a backup, you can take a basic backup or a full backup.

- **Basic backup.** Backs up only configuration files. You might want to perform this type of backup frequently, because the files it backs up change constantly. The files that are backed up are:

Directory	Sub-Directory or Files
/nsconfig/	• ns.conf • ZebOS.conf • rc.netscaler • snmpd.conf • nsbefore.sh • nsafter.sh • inetd.conf • ntp.conf • syslog.conf • newsyslog.conf • crontab • host.conf • hosts • ttys • sshd_config • httpd.conf

Directory	Sub-Directory or Files
	• monitrc • rc.conf • ssh_config • localtime • issue • issue.net
/var/	• download/* • log/wicmd.log • wi/tomcat/webapps/* • wi/tomcat/logs/* • wi/tomcat/conf/catalina/localhost/* • nslw.bin/etc/krb.conf • nslw.bin/etc/krb.keytab • netscaler/locdb/* • lib/likewise/db/* • vpn/bookmark/* • netscaler/crl • nstemplates/* • learnt_data/*
/netscaler/	• custom.html • vsr.htm

- **Full backup.** In addition to the files that are backed up by a basic backup, a full backup backs up some less frequently updated files. The files that are backed up when using the full backup option are:

Directory	Sub-Directory or Files
/nsconfig/	• ssl/* • license/* • fips/*
/var/	• netscaler/ssl/* • wi/java_home/jre/lib/security/cacerts/* • wi/java_home/lib/security/cacerts/*

The backup is stored as a compressed TAR file in the /var/ns_sys_backup/ directory. To avoid issues due to non-availability of disk space, you can store a maximum of 50 backup files in this directory. You can use the rm system backup command to delete existing backup files so that you can create more backups.

Note:

- While the backup operation is in progress, do not execute commands that affect the configuration.
- If a file that is required to be backed up is not available, the operation skips that file.

To backup the NetScaler by using the NetScaler command line interface

At the command prompt, do the following:

1. Save the NetScaler configurations.
   save ns config

2. Create the backup file.
   create system backup [<fileName>] -level <basic ¦ full> -comment <string>

   Note: If the file name is not specified, the appliance creates a TAR file with the following naming convention: backup_<level>_<nsip_address>_<date-timestamp>.tgz.
   Example: To backup the full appliance using the default naming convention for the backup file.
   ```
   > create system backup -level full
   ```
3. Verify that the backup file was created.
   show system backup

   You can view properties of a specific backup file by using the fileName parameter.

To backup the NetScaler by using the configuration utility

Navigate to System > Backup and Restore, click Backup and then specify the details of the backup.

Restoring the NetScaler Appliance

When you restore the appliance from a backup file, the restore operation untars the backup file into the /var/ns_sys_backup/ directory. Once the untar operation is complete, the files are copied to their respective directories.

Attention: The restore operation does not succeed if the backup file is renamed or if the contents of the file are modified.

To restore the NetScaler by using the command line interface

At the command prompt, do the following:

1. Obtain a list of the backup files available on the appliance.
   show system backup

2. Restore the appliance by specifying one of the backup files.
   restore system backup <filename>

   Example: To restore by using a full backup of an appliance.
   ```
   > restore system backup backup_full_<nsip_address>_<date-timestamp>.tgz
   ```
3. Reboot the appliance.
   reboot

To restore the NetScaler by using the configuration utility

Navigate to System > Backup and Restore, right-click the backup file to be restored and click Restore.

Restarting or Shutting down the Appliance

Sep 16, 2014

The NetScaler appliance can be remotely restarted or shut down from the available user interfaces. When a standalone NetScaler appliance is restarted or shut down, the unsaved configurations (configurations performed since the last save ns config command was issued) are lost.

In a high availability setup, when the primary appliance is rebooted/shut down, the secondary appliance takes over and becomes the primary. The unsaved configurations from the old primary are available on the new primary appliance.

You can also restart the appliance by only rebooting the NetScaler software and not rebooting the underlying operating system. This is called a warm reboot. For example, when you add a new license or change the NetScaler IP address, you can warm reboot the NetScaler appliance for these changes to take place.

Note: Warm reboot can be performed only on nCore appliances.

To restart the NetScaler by using the command line interface

At the command prompt, type:

reboot [-warm]

To restart the NetScaler by using the configuration utility

1. In the configuration utility, click Reboot on the home page of the Configuration tab.
2. When prompted to reboot, select Save configuration to make sure that you do not lose any configurations.
   Note: You can perform a warm reboot by selecting Warm reboot.

To shut down the NetScaler by using the command line interface

At the command prompt, type:

- shutdown –p now: Shuts down the software and switches off the NetScaler. To restart NetScaler MPX, press the AC power switch. To Restart NetScaler VPX , restart the VPX instance.
- shutdown –h now: Shuts down the software and leaves the NetScaler switched on. Press any key to restart the NetScaler. This command does not switch off the NetScaler. Therefore, do not switch off the AC power or remove the AC power cables.

Note: The appliance cannot be shut down from the configuration utility.

Authentication and Authorization

Mar 17, 2014

To configure NetScaler authentication and authorization, you must first define the users who have access to the NetScaler appliance, and then you can organize these users into groups. After configuring users and groups, you need to configure command policies to define types of access, and assign the policies to users and/or groups.

You must log on as an administrator to configure users, groups, and command policies. The default NetScaler administrator user name is *nsroot*. After logging on as the default administrator, you should change the password for the nsroot account. Once you have changed the password, no user can access the NetScaler appliance until you create an account for that user. If you forget the administrator password after changing it from the default, you can reset it to nsroot.

Configuring Users, User Groups, and Command Policies

Oct 06, 2016

You must define your users by configuring accounts for them. To simplify the management of user accounts, you can organize them into groups. You can create command policies, or use built-in command policies, to regulate user access to commands.

You can also customize the command-line prompt for a user. Prompts can be defined in a user's configuration, in a user-group configuration, and in the global system configuration settings. The prompt displayed for a given user is determined by the following order of precedence:

1. Display the prompt as defined in the user's configuration.
2. Display the prompt as defined in the group configuration for the user's group.
3. Display the prompt as defined in the system global configuration settings.

You can specify a timeout value for inactive CLI sessions for a system user. If a user's CLI session is idle for a time that exceeds the timeout value, the NetScaler appliance terminates the connection. The timeout can be defined in a user's configuration, in a user-group configuration, and in the global system configuration settings. The timeout for inactive CLI sessions for a user is determined by the following order of precedence:

1. Timeout value as defined in the user's configuration.
2. Timeout value as defined in the group configuration for the user's group.
3. Timeout value as defined in the global system configuration settings.

This document includes the following details:

- Configuring User Accounts
- Configuring User Groups
- Configuring Command Policies

Configuring User Accounts

Updated: 2014-08-07

To configure user accounts, you simply specify user names and passwords. You can change passwords and remove user accounts at any time.

To create a user account by using the command line interface

At the command prompt, type the following commands to create a user account and verify the configuration:

- add system user <userName> [-promptString <string>] [-timeout <secs>]
- show system user <userName>

Example

```
> add system user johnd -promptString user-%u-at-%T

Enter password:
Confirm password:
```

To configure a user account by using the configuration utility

Navigate to System > User Administration > Users, and create the user.

Configuring User Groups

Updated: 2014-08-07

After configuring a user group, you can easily grant the same access rights to everyone in the group. To configure a group, you create the group and bind users to the group. You can bind each user account to more than one group. Binding user accounts to multiple groups might allow more flexibility when applying command policies.

To create a user group by using the command line interface

At the command prompt, type the following commands to create a user group and verify the configuration:

- add system group <groupName> [-promptString <string>] [-timeout <secs>]
- show system group <groupName>

Example

```
> add system group Managers -promptString Group-Managers-at-%h
```

To bind a user to a group by using the command line interface

At the command prompt, type the following commands to bind a user account to a group and verify the configuration:

- bind system group <groupName> -userName <userName>
- show system group <groupName>

Example

```
> bind system group Managers -userName user1
```

To configure a user group by using the configuration utility

Navigate to System > User Administration > Groups, and create the user group.

Note: To add members to the group, in the Members section, click Add. Select users from the Available list and add them to the Configured list.

Configuring Command Policies

Command policies regulate which commands, command groups, virtual servers, and other entities that users and user groups are permitted to use.

The appliance provides a set of built-in command policies, and you can configure custom policies. To apply the policies, you bind them to users and/or groups.

Here are the key points to keep in mind when defining and applying command policies.

- You cannot create global command policies. Command policies must be bound directly to the users and groups on the appliance.
- Users or groups with no associated command policies are subject to the default (DENY-ALL) command policy, and are therefore unable to execute any configuration commands until the proper command policies are bound to their accounts.
- All users inherit the policies of the groups to which they belong.
- You must assign a priority to a command policy when you bind it to a user account or group account. This enables the

appliance to determine which policy has priority when two or more conflicting policies apply to the same user or group.

- The following commands are available by default to any user and are unaffected by any command you specify: help, show cli attribute, set cli prompt, clear cli prompt, show cli prompt, alias, unalias, history, quit, exit, whoami, config, set cli mode, unset cli mode, and show cli mode.

Built-in Command Policies

Updated: 2015-06-15

The following table describes the built-in policies.

Table 1. Built-in Command Policies

Policy name	Allows
read-only	Read-only access to all show commands except show ns runningConfig, show ns ns.conf, and the show commands for the NetScaler command group.
operator	Read-only access and access to commands to enable and disable services and servers.
network	Full access, except to the set and unset SSL commands, show ns ns.conf, show ns runningConfig, and show gslb runningConfig commands.
sysadmin	[From NetScaler 11 onwards] A sysadmin is lower than a superuser is terms of access allowed on the appliance. A sysadmin user can perform all NetScaler operations with the following exceptions: no access to the NetScaler shell, cannot perform user configurations, cannot perform partition configurations, and some other configurations as stated in the sysadmin command policy.
superuser	Full access. Same privileges as the nsroot user.

Creating Custom Command Policies

Updated: 2015-06-15

Regular expression support is offered for users with the resources to maintain more customized expressions, and for those deployments that require the flexibility that regular expressions offer. For most users, the built-in command policies are sufficient. Users who need additional levels of control but are unfamiliar with regular expressions may want to use only simple expressions, such as those in the examples provided in this section, to maintain policy readability.

When you use a regular expression to create a command policy, keep the following in mind.

- When you use regular expressions to define commands that will be affected by a command policy, you must enclose the commands in double quotation marks. For example, to create a command policy that includes all commands that begin with show, type:
  "^show .*$"

  To create a command policy that includes all commands that begin with rm, type:

"^rm .*$"

- Regular expressions used in command policies are not case sensitive.

The following table lists examples of regular expressions:

Table 2. Examples of Regular Expressions for Command Policies

Command specification	Matches these commands
"^rm\s+.*$"	All remove actions, because all remove actions begin with the rm string, followed by a space and additional parameters such as command groups, command object types, and arguments.
"^show\s+.*$"	All show commands, because all show actions begin with the show string, followed by a space and additional parameters such as command groups, command object types, and arguments.
"^shell$"	The shell command alone, but not combined with any additional parameters such as command groups, command object types, and arguments.
"^add\s+vserver\s+.*$"	All create virtual server actions, which consist of the add vserver command followed by a space and additional parameters such as command groups, command object types, and arguments.
"^add\s+(lb\s+vserver)\s+.*"	All create lb virtual server actions, which consist of the add lb vserver command followed by a space and additional parameters such as command groups, command object types, and arguments.

The following table shows the command specifications for each of the built-in command policies.

Table 3. Expressions Used in the Built-in Command Policies

Policy name	Command specification regular expression
read-only	(^man.*)|(^show\s+(?!system)(?!configstatus)(?!ns ns\.conf)(?!ns savedconfig)(?!ns runningConfig)(?!gslb runningConfig)(?!audit messages)(?!techsupport).*)|(^stat.*)
operator	(^man.*)|(^show\s+(?!system)(?!configstatus)(?!ns ns\.conf)(?!ns savedconfig)(?!ns runningConfig)(?!gslb runningConfig)(?!audit messages)(?!techsupport).*)|(^stat.*)|(^(enable|disable) (server|service).*)
network	^(?!clear ns config.*)(?!scp.*)(?!set ssl fips)(?!reset ssl fips)(?!diff ns config)(?!shell)(?!reboot)(?!batch)\S+\s+(?!system)(?!configstatus)(?!ns ns\.conf)(?!ns savedconfig)(?!ns runningConfig)(?!gslb runningConfig)(?!techsupport).*
sysadmin	[From NetScaler 11 onwards] ^(?!shell)(?!sftp)(?!scp)(?!batch)(?!source)(?!.*superuser)(?!.*nsroot)(?!show\s+system\s+(user|cmdPolicy))(?!(set|add|rm|create|export|kill)\s+system)(?!(unbind|bind)\s+system\s+(user|group))(?!diff\s+ns\s+config)(?!\S+\s+ns\s+partition).*

Policy name	Command specification regular expression
superuser	.*

To create a command policy by using the command line interface

At the command prompt, type the following commands to create a command policy and verify the configuration:

- add system cmdPolicy <policyname> <action> <cmdspec>
- show system cmdPolicy <policyName>

Example

```
> add system cmdPolicy read_all ALLOW (^show\s+(!system)(!ns ns.conf)(!ns runningConfig).*)|(^stat.*)
```

To configure a command policy by using the configuration utility

Navigate to System > User Administration > Command Policies, and create the command policy.

Binding Command Policies to Users and Groups

Updated: 2014-08-07

Once you have defined your command policies, you must bind them to the appropriate user accounts and groups. When you bind a policy, you must assign it a priority so that the appliance can determine which command policy to follow when two or more applicable command policies are in conflict.

Command policies are evaluated in the following order:

- Command policies bound directly to users and the corresponding groups are evaluated according to priority number. A command policy with a lower priority number is evaluated before one with a higher priority number. Therefore, any privileges the lower-numbered command policy explicitly grants or denies are not overridden by a higher-numbered command policy.
- When two command policies, one bound to a user account and other to a group, have the same priority number, the command policy bound directly to the user account is evaluated first.

To bind command policies to a user by using the command line interface

At the command prompt, type the following commands to bind a command policy to a user and verify the configuration:

- bind system user <userName> -policyName <policyName> <priority>
- show system user <userName>

Example

```
> bind system user user1 -policyName read_all 1
```

To bind command policies to a user by using the configuration utility

Navigate to System > User Administration > Users, select the user and bind command policies.
Optionally, you can modify the default priority to ensure that the policy is evaluated in the proper order.

To bind command policies to a group by using the command line interface

At the command prompt, type the following commands to bind a command policy to a group, and verify the configuration:

- bind system group <groupName> -policyName <policyName> <priority>
- show system group <groupName>

Example

```
> bind system group Managers -policyName read_all 1
```

To bind command policies to a group by using the configuration utility

Navigate to System > User Administration > Groups, select the group, and bind command policies.
Optionally, you can modify the default priority to ensure that the policy is evaluated in the proper order.

Resetting the Default Administrator (nsroot) Password

Nov 24, 2014

The nsroot account provides complete access to all features of the appliance. Therefore, to preserve security, the nsroot account should be used only when necessary, and only individuals whose duties require full access should know the password for the nsroot account. Frequently changing the nsroot password is advisable. If you lose the password, you can reset it to the default and then change it.

To reset the nsroot password, you must boot the appliance into single user mode, mount the file systems in read/write mode, and remove the set NetScaler user nsroot entry from the ns.conf file. You can then reboot, log on with the default password, and choose a new password.

To reset the nsroot password

1. Connect a computer to the console port of the NetScaler ADC and log on.
   Note: You cannot log on by using SSH to perform this procedure; you must connect directly to the appliance.
2. Reboot the NetScaler ADC.
3. Press CTRL+C when the following message appears:
   `Press [Ctrl-C] for command prompt, or any other key to boot immediately.`

   `Booting [kernel] in # seconds.`
4. Run the following command to start the NetScaler in a single user mode:
   `boot -s`

   Note: If boot -s does not work, then try reboot -- -s and appliance will reboot in single user mode.
   After the appliance boots, it displays the following message:

   `Enter full path name of shell or RETURN for /bin/sh:`
5. Press ENTER key to display the # prompt, and type the following commands to mount the file systems:
   1. Run the following command to check the disk consistency:
      `fsck /dev/ad0s1a`

      Note: Your flash drive will have a specific device name depending on your NetScaler; hence, you have to replace ad0s1a in the preceding command with the appropriate device name.
   2. Run the following command to display the mounted partitions:
      `df`

      If the flash partition is not listed, you need to mount it manually.
   3. Run the following command to mount the flash drive:
      `mount /dev/ad0s1a /flash`
6. Run the following command to change to the nsconfig directory:
   `cd /flash/nsconfig`
7. Run the following commands to rewrite the **ns.conf** file and remove the set of system commands defaulting to the nsroot user:
   1. Run the following command to create a new configuration file that does not have commands defaulting to the

nsroot user:
grep –v “set system user nsroot” ns.conf > new.conf

2. Run the following command to make a backup of the existing configuration file:
   mv ns.conf old.ns.conf

3. Run the following command to rename the new.conf file to ns.conf:
   mv new.conf ns.conf

8. Run the following command to reboot the NetScaler:
   reboot

9. Log on using the default nsroot user credentials.
10. Run the following command to reset the nsroot user password:
    set system user nsroot <New_Password>

Example of a User Scenario

Sep 06, 2013

The following example shows how to create a complete set of user accounts, groups, and command policies and bind each policy to the appropriate groups and users. The company, Example Manufacturing, Inc., has three users who can access the NetScaler appliance:

- **John Doe.** The IT manager. John needs to be able to see all parts of the NetScaler configuration but does not need to modify anything.
- **Maria Ramiez.** The lead IT administrator. Maria needs to be able to see and modify all parts of the NetScaler configuration except for NetScaler commands (which local policy dictates must be performed while logged on as nsroot).
- **Michael Baldrock.** The IT administrator in charge of load balancing. Michael needs to be able to see all parts of the NetScaler configuration, but needs to modify only the load balancing functions.

The following table shows the breakdown of network information, user account names, group names, and command policies for the sample company.

Table 1. Sample Values for Creating Entities

Field	Value	Note
NetScaler host name	ns01.example.net	N/A
User accounts	johnd, mariar, and michaelb	John Doe, IT manager, Maria Ramirez, IT administrator and Michael Baldrock, IT administrator.
Groups	Managers and SysOps	All managers and all IT administrators.
Command Policies	read_all, modify_lb, and modify_all	Allow complete read-only access, Allow modify access to load balancing, and Allow complete modify access.

The following description walks you through the process of creating a complete set of user accounts, groups, and command policies on the NetScaler appliance named ns01.example.net.

The description includes procedures for binding the appropriate user accounts and groups to one another, and binding appropriate command policies to the user accounts and groups.

This example illustrates how you can use prioritization to grant precise access and privileges to each user in the IT department.

The example assumes that initial installation and configuration have already been performed on the NetScaler.

Configuration steps

1. Use the procedure described in "Configuring User Accounts" to create user accounts **johnd**, **mariar**, and **michaelb**.
2. Use the procedure described in "Configuring User Groups" to create user groups **Managers** and **SysOps**, and then bind the users **mariar** and **michaelb** to the **SysOps** group and the user **johnd** to the **Managers** group.

3. Use the procedure described in "Creating Custom Command Policies" to create the following command policies:
   - **read_all** with action **Allow** and command spec "(^show\s+(?!system)(?!ns ns.conf)(?!ns runningConfig).*)|(^stat.*)"
   - **modify_lb** with action as **Allow** and the command spec "^set\s+lb\s+.*$"
   - **modify_all** with action as **Allow** and the command spec **"^\S+\s+(?!system).*"**
4. Use the procedure described in "Binding Command Policies to Users and Groups" to bind the **read_all** command policy to the **SysOps** group, with priority value **1**.
5. Use the procedure described in "Binding Command Policies to Users and Groups" to bind the **modify_lb** command policy to user **michaelb**, with priority value **5**.

The configuration you just created results in the following:

- John Doe, the IT manager, has read-only access to the entire NetScaler configuration, but he cannot make modifications.
- Maria Ramirez, the IT lead, has near-complete access to all areas of the NetScaler configuration, having to log on only to perform NetScaler-level commands.
- Michael Baldrock, the IT administrator responsible for load balancing, has read-only access to the NetScaler configuration, and can modify the configuration options for load balancing.

The set of command policies that applies to a specific user is a combination of command policies applied directly to the user's account and command policies applied to the group(s) of which the user is a member.

Each time a user enters a command, the operating system searches the command policies for that user until it finds a policy with an ALLOW or DENY action that matches the command. When it finds a match, the operating system stops its command policy search and allows or denies access to the command.

If the operating system finds no matching command policy, it denies the user access to the command, in accordance with the NetScaler appliance's default deny policy.

Note: When placing a user into multiple groups, take care not to cause unintended user command restrictions or privileges. To avoid these conflicts, when organizing your users in groups, bear in mind the NetScaler command policy search procedure and policy ordering rules.

Configuring External User Authentication

Jun 01, 2015

External user authentication is the process of authenticating the users of the Citrix NetScaler appliance by using an external authentication server. The NetScaler supports LDAP, RADIUS, and TACACS+ authentication servers. To configure external user authentication, you must create authentication policies. You can configure one or many authentication policies, depending on your authentication needs. An authentication policy consists of an expression and an action. Authentication policies use NetScaler classic expressions, which are described in detail in "Policy Configuration and Reference."

After creating an authentication policy, you bind it to the system global entity and assign a priority to it. You can create simple server configurations by binding a single authentication policy to the system global entity. Or, you can configure a cascade of authentication servers by binding multiple policies to the system global entity. If no authentication policies are bound to the system, users are authenticated by the onboard system.

This document includes the following details:

- Configuring LDAP Authentication
- Configuring RADIUS Authentication
- Configuring TACACS+ Authentication
- Binding the Authentication Policies to the System Global Entity

Configuring LDAP Authentication

Updated: 2014-12-29

You can configure the NetScaler appliance to authenticate user access with one or more LDAP servers. LDAP authorization requires identical group names in Active Directory, on the LDAP server, and on the appliance. The characters and case must also be the same.

By default, LDAP authentication is secured by using SSL/TLS protocol. There are two types of secure LDAP connections. In the first type, the LDAP server accepts the SSL/TLS connection on a port separate from the port used to accept clear LDAP connections. After users establish the SSL/TLS connection, LDAP traffic can be sent over the connection. The second type allows both unsecure and secure LDAP connections and is handled by a single port on the server. In this scenario, to create a secure connection, the client first establishes a clear LDAP connection. Then the LDAP command StartTLS is sent to the server over the connection. If the LDAP server supports StartTLS, the connection is converted to a secure LDAP connection by using TLS.

The port numbers for LDAP connections are:

- 389 for unsecured LDAP connections
- 636 for secure LDAP connections
- 3268 for Microsoft unsecure LDAP connections
- 3269 for Microsoft secure LDAP connections

LDAP connections that use the StartTLS command use port number 389. If port numbers 389 or 3268 are configured on the appliance, it tries to use StartTLS to make the connection. If any other port number is used, connection attempts use SSL/TLS. If StartTLS or SSL/TLS cannot be used, the connection fails.

When configuring the LDAP server, the case of the alphabetic characters must match that on the server and on the appliance. If the root directory of the LDAP server is specified, all of the subdirectories are also searched to find the user attribute. In large directories, this can affect performance. For this reason, Citrix recommends that you use a specific organizational unit (OU).

The following table lists examples of user attribute fields for LDAP servers.

Table 1. User Attribute Fields for LDAP Servers

LDAP server	User attribute	Case sensitive?
Microsoft Active Directory	Server sAMAccountName	No
Novell eDirectory	cn	Yes
IBM Directory Server	uid	Yes
Lotus Domino	CN	Yes
Sun ONE directory (formerly iPlanet)	uid or cn	Yes

The following table lists examples of the base distinguished name (DN).

Table 2. Examples of Base Distinguished Name

LDAP server	Base DN
Microsoft Active Directory	DC=citrix, DC=local
Novell eDirectory	dc=citrix, dc=net
IBM Directory Server	cn=users
Lotus Domino	OU=City, O=Citrix, C=US

LDAP server	Base DN
Sun ONE directory (formerly iPlanet)	ou=People, dc=citrix, dc=com

The following table lists examples of the bind distinguished name (DN).

Table 3. Examples of Bind Distinguished Name

LDAP server	Bind DN
Microsoft Active Directory	CN=Administrator, CN=Users, DC=citrix, DC=local
Novell eDirectory	cn=admin, dc=citrix, dc=net
IBM Directory Server	LDAP_dn
Lotus Domino	CN=Notes Administrator, O=Citrix, C=US
Sun ONE directory (formerly iPlanet)	uid=admin, ou=Administrators, ou=TopologyManagement, o=NetscapeRoot

To configure LDAP authentication by using the command line interface

At the command prompt, do the following:

1. Create an LDAP action.
   add authentication ldapAction <name> {-serverIP <ip_addr|ipv6_addr|*> | {-serverName <string>}} >] [-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-ldapLoginName <string>] [-groupAttrName <string>] [-subAttributeName <string>]

   Example:
   add authentication ldapAction ldap70 -serverIP <IP> -authTimeout 30 -ldapBase "CN=xxxxx,DC=xxxx,DC=xxx" -ldapBindDn "CN=xxxxx,CN=xxxxx,DC=xxxx,DC=xxx" -ld
2. Create an LDAP policy.
   add authentication ldapPolicy <name> <rule> [<reqAction>]

   Example:
   add authentication ldappolicy ldap_pol ns_true ldap70
3. Bind the LDAP policy to the following bind points at which the policy will be evaluated.
   - **System Global**: bind system global <policyName> [-priority <positive_integer>]
   - **VPN Global**: bind vpn global <policyName> [-priority <positive_integer>]
   - **Authentication Server**: bind authentication vserver <name> [-policy <string> [-priority <positive_integer>]
   - **VPN Server**: bind vpn vserver <name> [-policy <string> [-priority <positive_integer>]

To configure LDAP authentication by using the configuration utility

Navigate to System > Authentication > LDAP, and create the LDAP authentication policy.

Determining attributes in the LDAP directory

If you need help determining your LDAP directory attributes, you can easily look them up with the free LDAP browser from Softerra.

You can download the LDAP browser from the Softerra LDAP Administrator Web site at http://www.ldapbrowser.com. After the browser is installed, set the following attributes:

- The host name or IP address of your LDAP server.
- The port of your LDAP server. The default is 389.
- The base DN field can be left blank.
- The information provided by the LDAP browser can help you determine the base DN needed for the Authentication tab.
- The Anonymous Bind check determines whether the LDAP server requires user credentials for the browser to connect to it. If the LDAP server requires credentials, leave the check box cleared.

After completing the settings, the LDAP browser displays the profile name in the left pane and connects to the LDAP server.

Configuring RADIUS Authentication

Updated: 2014-08-08

You can configure the NetScaler appliance to authenticate user access with one or more RADIUS servers. If you are using RSA SecurID, SafeWord, or Gemalto Protiva products, use a RADIUS server.

Your configuration might require using a network access server IP address (NAS IP) or a network access server identifier (NAS ID). When configuring the appliance to use a RADIUS authentication server, use the following guidelines:

- If you enable use of the NAS IP, the appliance sends its configured IP address to the RADIUS server, rather than the source IP address used in establishing the RADIUS connection.
- If you configure the NAS ID, the appliance sends the identifier to the RADIUS server. If you do not configure the NAS ID, the appliance sends its host name to the RADIUS server.
- When the NAS IP is enabled, the appliance ignores any NAS ID that was configured by using the NAS IP to communicate with the RADIUS server.

To configure RADIUS authentication by using the configuration utility

Navigate to System > Authentication > Radius, and create the RADIUS authentication policy.

Choosing RADIUS authentication protocols

The NetScaler appliance supports implementations of RADIUS that are configured to use any of several protocols for user authentication, including:

- Password Authentication Protocol
- Challenge-Handshake Authentication Protocol (CHAP)

- Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP Version 1 and Version 2)

If your deployment of the appliance is configured to use RADIUS authentication and your RADIUS server is configured to use Password Authentication Protocol, you can strengthen user authentication by assigning a strong shared secret to the RADIUS server. Strong RADIUS shared secrets consist of random sequences of uppercase and lowercase letters, numbers, and punctuation, and are at least 22 characters long. If possible, use a random character generation program to determine RADIUS shared secrets.

To further protect RADIUS traffic, assign a different shared secret to each appliance or virtual server. When you define clients on the RADIUS server, you can also assign a separate shared secret to each client. If you do this, you must configure separately each policy that uses RADIUS authentication.

Shared secrets are configured on the appliance when a RADIUS policy is created.

Configuring IP address extraction

You can configure the appliance to extract the IP address from a RADIUS server. When a user authenticates with the RADIUS server, the server returns a framed IP address that is assigned to the user. The following are attributes for IP address extraction:
- Allows a remote RADIUS server to supply an IP address from the internal network for a user logged on to the appliance.
- Allows configuration for any RADIUS attribute using the type ipaddress, including those that are vendor encoded.

When configuring the RADIUS server for IP address extraction, you configure the vendor identifier and the attribute type.

The vendor identifier enables the RADIUS server to assign an IP address to the client from a pool of IP addresses that are configured on the RADIUS server. The vendor ID and attributes are used to make the association between the RADIUS client and the RADIUS server. The vendor ID is the attribute in the RADIUS response that provides the IP address of the internal network. A value of zero indicates that the attribute is not vendor encoded. The attribute type is the remote IP address attribute in a RADIUS response. The minimum value is one and the maximum value is 255.

A common configuration is to extract the RADIUS attribute *framed IP address*. The vendor ID is set to zero or is not specified. The attribute type is set to eight.

To configure IP address extraction by using the configuration utility

1. Navigate to System > Authentication > Radius, and select a policy.
2. Modify the server parameters and set relevant values in Group Vendor Identifier and Group Attribute Type fields.

Configuring TACACS+ Authentication

Updated: 2014-08-07

You can configure a TACACS+ server for authentication. Similar to RADIUS authentication, TACACS+ uses a secret key, an IP address, and the port number. The default port number is 49. To configure the appliance to use a TACACS+ server, provide the server IP address and the TACACS+ secret. The port needs to be specified only when the server port number in use is something other than the default port number of 49.

To configure TACACS+ authentication by using the configuration utility

Navigate to System > Authentication > TACACS, and create the TACACS authentication policy.
After the TACACS+ server settings are configured on the appliance, bind the policy to the system global entity. For more information about binding authentication policies globally, see "Binding the Authentication Policies to the System Global Entity."

Binding the Authentication Policies to the System Global Entity

Updated: 2014-12-30

When the authentication policies are configured, bind the policies to the system global entity.

To bind an authentication policy to system global using the command line interface

At the command line prompt, do the following:

bind system global <policyName> [-priority <positive_integer>]

Example:
```
bind system global ldappol1 -priority 10
```

To bind an authentication policy to system global using the configuration utility

1. Navigate to System > Authentication, and select the required authentication type.
2. On the Policies tab, click Action > Global Bindings.
3. Click Insert Policy and under Policy Name, select the policy and click OK.

TCP Configurations

Nov 05, 2015

TCP configurations for a NetScaler appliance can be specified in an entity called a TCP profile, which is a collection of TCP settings. The TCP profile can then be associated with services or virtual servers that want to use these TCP configurations.

A default TCP profile can be configured to set the TCP configurations that will be applied by default, globally to all services and virtual servers.

Note: When a TCP parameter has different values for service, virtual server, and globally, the value of the most-specific entity (the service) is given the highest precedence.
The NetScaler appliance also provides other approaches for configuring TCP. Read on for more information.

The NetScaler appliance supports the following TCP capabilities:

- Congestion control using New-Reno, and TCP Westwood algorithms.
- Window scaling to increase the TCP receive window size beyond its maximum value of 65,535 bytes.
  Note: Before configuring window scaling, make sure that:
  - You do not set a high value for the scale factor, because this could have adverse effects on the appliance and the network.
  - You do not configure window scaling unless you clearly know why you want to change the window size.
  - Both hosts in the TCP connection send a window scale option during connection establishment. If only one side of a connection sets this option, window scaling is not used for the connection.
  - Each connection for same session is an independent window scaling session. For example, when a client's request and the server's response flow through the appliance, it is possible to have window scaling between the client and the appliance without window scaling between the appliance and the server.
- TCP maximum congestion window size that is user configurable. The default value is 8190 bytes.
- Selective acknowledgment (SACK), using which the data receiver (either a NetScaler appliance or a client) notifies the sender about all the segments that have been received successfully.
- Forward acknowledgment (FACK) avoids TCP congestion by explicitly measuring the total number of data bytes outstanding in the network, and helping the sender (either a NetScaler ADC or a client) control the amount of data injected into the network during retransmission timeouts.
- TCP connection multiplexing enables reuse of existing TCP connections. The NetScaler appliance stores established TCP connections to the reuse pool. Whenever a client request is received, appliance checks for an available connection in the reuse pool and serves the new client if the connection is available. If it is unavailable, the appliance creates a new connection for the client request and stores the connection to the reuse pool.
  NetScaler supports connection multiplexing for HTTP, SSL, and DataStream connection types.
- Dynamic receive buffering allows the receive buffer to be adjusted dynamically based on memory and network conditions.
- MPTCP connections between client and NetScaler. MPTCP connections are not supported between NetScaler and the backend server.
  The NetScaler implementation of MPTCP is RFC 6824 compliant.

  Note:
  - For MPTCP to work, both sides of the connection (client and server) must support it. If you use the NetScaler appliance as an MPTCP gateway for your servers, the servers do not have to support MPTCP.
  - The NetScaler appliance does not initiate subflows (MP_JOIN's). The appliance expects the client to initiate subflows.
- TCP keep-alive to monitor the TCP connections to verify if the peers are up.
- Extracting TCP/IP path overlay and inserting client-IP HTTP header. Data transport through overlay networks often uses connection termination or Network Address Translation (NAT), in which the IP address of the source client is lost. To avoid this, the Netscaler appliance extracts the TCP/IP path overlay option and inserts the source client's IP address into the HTTP header. With the IP address in the header, the web server can identify the source client that made the connection. The extracted data is valid for lifetime of the TCP connection and therefore, this prevents the next hop host from having to interpret the option again. This option is applicable only for web services that have the client-IP insertion option enabled.

Additionally, NetScaler provides configuration support for the following:

- Synchronizing cookie for TCP handshake with clients. Disabling this capability prevents SYN attack protection on the NetScaler appliance.
- Learning MSS to enable MSS learning for all the virtual servers configured on the appliance.

This document includes the following details:

Setting Global TCP Parameters

Updated: 2014-08-08

The NetScaler appliance allows you to specify values for TCP parameters that are applicable to all NetScaler services and virtual servers. This can be done using:

- Default TCP profile
- Global TCP command
- TCP buffering feature

Note: The recvBuffSize parameter of the set ns tcpParam command is deprecated from release 9.2 onwards. In later releases, set the buffer size by using the bufferSize parameter of the set ns tcpProfile command. If you upgrade to a release where the recvBuffSize parameter is deprecated, the bufferSize parameter is set to its default value.

Default TCP profile

A TCP profile, named as **nstcp_default_profile**, is used to specify TCP configurations that will used if no TCP configurations are provided at the service or virtual server level.

Note:
- Not all TCP parameters can be configured through the default TCP profile. Some settings have to be performed by using the global TCP command (see section below).
- The default profile does not have to be explicitly bound to a service or virtual server.

To configure the default TCP profile

- Using the command line interface, at the command prompt enter:

  set ns tcpProfile nstcp_default_profile ...

- On the configuration utility, navigate to System > Profiles, click TCP Profiles and update nstcp_default_profile.

Global TCP command

Another approach you can use to configure global TCP parameters is the global TCP command. In addition to some unique parameters, this command duplicates some parameters that can be set by using a TCP profile. Any update made to these duplicate parameters is reflected in the corresponding parameter in the default TCP profile.

For example, if the SACK parameter is updated using this approach, the value is reflected in the SACK parameter of the default TCP profile (nstcp_default_profile).

Note: Citrix recommends that you use this approach only for TCP parameters that are not available in the default TCP profile.
To configure the global TCP command

- Using the command line interface, at the command prompt enter:

  set ns tcpParam ...

- On the configuration utility, navigate to System > Settings, click Change TCP parameters and update the required TCP parameters.

TCP buffering feature

NetScaler provides a feature called TCP buffering that you can use to specify the TCP buffer size. The feature can be enabled globally or at service level.

Note: The buffer size can also be configured in the default TCP profile. If the buffer size has different values in the TCP buffering feature and the default TCP profile, the greater value is applied.
To configure the TCP buffering feature globally

- At the command prompt enter:

  enable ns mode TCPB

  set ns tcpbufParam -size <positiveInteger> -memLimit <positiveInteger>

- On the configuration utility, navigate to System > Settings, click Configure Modes and select TCP Buffering.
  And, navigate to System > Settings, click Change TCP parameters and specify the values for Buffer size and Memory usage limit.

Setting Service or Virtual Server Specific TCP Parameters

Updated: 2014-08-08

Using TCP profiles, you can specify TCP parameters for services and virtual servers. You must define a TCP profile (or use a built-in TCP profile) and associate the profile with the appropriate service and virtual server.

Note:
- You can also modify the TCP parameters of default profiles as per your requirements. For more information on built-in TCP profiles, see Built-in TCP Profiles.

- You can specify the TCP buffer size at service level using the parameters specified by the TCP buffering feature. For more information, see TCP buffering feature.

To specify service or virtual server level TCP configurations by using the command line interface

At the command prompt, perform the following:

1. Configure the TCP profile.
   set ns tcpProfile <profile-name>...
2. Bind the TCP profile to the service or virtual server.
   To bind the TCP profile to the service:

   set service <name> ....

   Example:
   `> set service service1 -tcpProfileName profile1`
   To bind the TCP profile to the virtual server:

   set lb vserver <name> ....

   Example:
   `> set lb vserver lbvserver1 -tcpProfileName profile1`

To specify service or virtual server level TCP configurations by using the configuration utility

At the configuration utility, perform the following:

1. Configure the TCP profile.
   Navigate to System > Profiles > TCP Profiles, and create the TCP profile.
2. Bind the TCP profile to the service or virtual server.
   Navigate to Traffic Management > Load Balancing > Services/Virtual Servers, and create the TCP profile, which should be bound to the service or virtual server.

Built-in TCP Profiles

Updated: 2014-04-07

For convenience of configuration, the NetScaler provides some built-in TCP profiles. Review the built-in profiles listed below and select a profile and use it as it is or modify it to meet your requirements. You can bind these profiles to your required services or virtual servers.

Table 1. Built-in TCP Profiles

Built-in profile	Description
nstcp_default_profile	Represents the default global TCP settings on the appliance.
nstcp_default_tcp_lan	Useful for back-end server connections, where these servers reside on the same LAN as the appliance.
nstcp_default_tcp_lan_thin_stream	Similar to the nstcp_default_tcp_lan profile; however, the settings are tuned to small size packet flows.
nstcp_default_tcp_interactive_stream	Similar to the nstcp_default_tcp_lan profile; however, it has a reduced delayed ACK timer and ACK on PUSH packet settings.
nstcp_default_tcp_lfp	Useful for long fat pipe networks (WAN) on the client side. Long fat pipe networks have long delay, high bandwidth lines with minimal packet drops.
nstcp_default_tcp_lfp_thin_stream	Similar to the nstcp_default_tcp_lfp profile; however, the settings are tuned for small size packet flows.
nstcp_default_tcp_lnp	Useful for long narrow pipe networks (WAN) on the client side. Long narrow pipe networks have considerable packet loss once in a while.

Built-in profile	Description
nstcp_default_tcp_lnp_thin_stream	Similar to the nstcp_default_tcp_lnp profile; however, the settings are tuned for small size packet flows.
nstcp_internal_apps	Useful for internal applications on the appliance (for example, GSLB sitesyncing). This contains tuned window scaling and SACK options for the desired applications. This profile should not be bound to applications other than internal applications.
nstcp_default_Mobile_profile	Useful for mobile devices.
nstcp_default_XA_XD_profile	Useful for a XenApp or XenDesktop deployment.

Sample TCP Configurations

Updated: 2015-04-28

Sample command line interface examples for configuring the following:

- Selective ACKnowledgment (SACK)
- Window Scaling (WS)
- Maximum Segment Size (MSS)
- NetScaler to learn the MSS of a virtual server
- TCP keep-alive
- Buffer size - using TCP profile
- Buffer size - using TCP buffering feature
- MPTCP
- Congestion control
- Dynamic receive buffering

Selective ACKnowledgment (SACK)

Enable SACK on the required TCP profile.

```
> set ns tcpProfile profile1 -SACK ENABLED
 Done
> set lb vserver lbvserver1 -tcpProfileName profile1
 Done
```

Window Scaling (WS)

Enable window scaling and set the window scaling factor on the required TCP profile.

```
> set ns tcpProfile profile1 –WS ENABLED –WSVal 9
 Done
> set lb vserver lbvserver1 -tcpProfileName profile1
 Done
```

Maximum Segment Size (MSS)

Update the MSS related configurations.

```
> set ns tcpProfile profile1 –mss 1460 - maxPktPerMss 512
 Done
> set lb vserver lbvserver1 -tcpProfileName profile1
 Done
```

NetScaler to learn the MSS of a virtual server

Enable the NetScaler to learn the VSS and update other related configurations.

```
> set ns tcpParam -learnVsvrMSS ENABLED –mssLearnInterval 180 -mssLearnDelay 3600
 Done
```

TCP keep-alive

Enable TCP keep-alive and update other related configurations.

```
> set ns tcpProfile profile1 –KA ENABLED –KaprobeUpdateLastactivity ENABLED -KAconnIdleTime 900 -KAmaxProbes 3 -KaprobeInterval 75
```

```
 Done
> set lb vserver lbvserver1 -tcpProfileName profile1
 Done
```

Buffer size - using TCP profile

Specify the buffer size.

```
> set ns tcpProfile profile1 –bufferSize 8190
 Done
> set lb vserver lbvserver1 -tcpProfileName profile1
 Done
```

Buffer size - using TCP buffering feature

Enable the TCP buffering feature (globally or for a service) and then specify the buffer size and the memory limit.

```
> enable ns feature TCPB
 Done
> set ns tcpbufParam -size 64 -memLimit 64
 Done
```

MPTCP

Enable MPTCP and then set the optional MPTCP configurations.

```
> set ns tcpProfile profile1 -mptcp ENABLED
 Done
> set ns tcpparam -mptcpConCloseOnPassiveSF ENABLED -mptcpChecksum ENABLED -mptcpSFtimeout 0 -mptcpSFReplaceTimeout 10
-mptcpMaxSF 4 -mptcpMaxPendingSF 4 -mptcpPendingJoinThreshold 0 -mptcpRTOsToSwitchSF 2 -mptcpUseBackupOnDSS ENABLED
 Done
```

Congestion control

Set the required TCP congestion control algorithm.

```
> set ns tcpProfile profile1 -flavor Westwood
Done
> set lb vserver lbvserver1 -tcpProfileName profile1
Done
```

Dynamic receive buffering

Enable dynamic receive buffering on the required TCP profile.

```
> set ns tcpProfile profile1 -dynamicReceiveBuffering ENABLED
Done
> set lb vserver lbvserver1 -tcpProfileName profile1
Done
```

HTTP Configurations

Jun 03, 2015

HTTP configurations for a NetScaler appliance can be specified in an entity called an HTTP profile, which is a collection of HTTP settings. The HTTP profile can then be associated with services or virtual servers that want to use these HTTP configurations.

A default HTTP profile can be configured to set the HTTP configurations that will be applied by default, globally to all services and virtual servers.

Note: When a HTTP parameter has different values for service, virtual server, and globally, the value of the most-specific entity (the service) is given the highest precedence.
The NetScaler appliance also provides other approaches for configuring HTTP. Read on for more information.

The NetScaler supports the following HTTP capabilities:

- WebSocket protocol which allows browsers and other clients to create a bi-directional, full duplex TCP connection to the servers. The NetScaler implementation of WebSocket is RFC 6455 complaint.
- SPDY (Speedy). For more information, see SPDY.

This document includes the following details:

- Setting Global HTTP Parameters
- Setting Service or Virtual Server Specific HTTP Parameters
- Built-in HTTP Profiles
- Sample HTTP Configurations

Setting Global HTTP Parameters

Updated: 2014-08-06

The NetScaler appliance allows you to specify values for HTTP parameters that are applicable to all NetScaler services and virtual servers. This can be done using:

- Default HTTP profile
- Global HTTP command

Default HTTP profile

A HTTP profile, named as **nshttp_default_profile**, is used to specify HTTP configurations that will be used if no HTTP configurations are provided at the service or virtual server level.

Note:

- Not all HTTP parameters can be configured through the default HTTP profile. Some settings have to be performed by using the global HTTP command (see section below).
- The default profile does not have to be explicitly bound to a service or virtual server.

To configure the default HTTP profile

- Using the command line interface, at the command prompt enter:

  set ns httpProfile nshttp_default_profile ...

- On the configuration utility, navigate to System > Profiles, click HTTP Profiles and update nshttp_default_profile.

Global HTTP command

Another approach you can use to configure global HTTP parameters is the global HTTP command. In addition to some unique parameters, this command duplicates some parameters that can be set by using a HTTP profile. Any update made to these duplicate parameters is reflected in the corresponding parameter in the default HTTP profile.

For example, if the maxReusePool parameter is updated using this approach, the value is reflected in the maxReusePool parameter of the default HTTP profile (nshttp_default_profile).

Note: Citrix recommends that you use this approach only for HTTP parameters that are not available in the default HTTP profile.
To configure the global HTTP command

- Using the command line interface, at the command prompt enter:

  set ns httpParam ...

- On the configuration utility, navigate to System > Settings, click Change HTTP parameters and update the required HTTP parameters.

Setting Service or Virtual Server Specific HTTP Parameters

Updated: 2014-08-07

Using HTTP profiles, you can specify HTTP parameters for services and virtual servers. You must define a HTTP profile (or use a built-in HTTP profile) and associate the profile with the appropriate service and virtual server.

Note: You can also modify the HTTP parameters of default profiles as per your requirements. For more information on built-in HTTP profiles, see Built-in HTTP Profiles.

To specify service or virtual server level HTTP configurations by using the command line interface

At the command prompt, perform the following:

1. Configure the HTTP profile.
   set ns httpProfile <profile-name>...

2. Bind the HTTP profile to the service or virtual server.
   To bind the HTTP profile to the service:

   set service <name> .....

   Example:
   ```
   > set service service1 -httpProfileName profile1
   ```
   To bind the HTTP profile to the virtual server:

   set lb vserver <name> .....

   Example:
   ```
   > set lb vserver lbvserver1 -httpProfileName profile1
   ```

To specify service or virtual server level HTTP configurations by using the configuration utility

At the configuration utility, perform the following:

1. Configure the HTTP profile.
   Navigate to System > Profiles > HTTP Profiles, and create the HTTP profile.

2. Bind the HTTP profile to the service or virtual server.
   Navigate to Traffic Management > Load Balancing > Services/Virtual Servers, and create the HTTP profile, which should be bound to the service/virtual server.

Built-in HTTP Profiles

Updated: 2014-04-25

For convenience of configuration, the NetScaler provides some built-in HTTP profiles. Review the profiles listed below and use it as it is or modify it to meet your requirements. You can bind these profiles to the required services or virtual servers.

Table 1. Built-in HTTP Profiles

Built-in profile	Description
nshttp_default_profile	Represents the default global HTTP settings on the appliance.
nshttp_default_strict_validation	Settings for deployments that require strict validation of HTTP requests and responses.

Sample HTTP Configurations

Updated: 2014-04-25

Sample command line interface examples to configure the following:

- HTTP band statistics
- WebSocket connections

HTTP band statistics

Specify the band size for HTTP requests and responses.

```
> set protocol httpBand reqBandSize 300 respBandSize 2048
Done
> show protocol httpband -type REQUEST
```

WebSocket connections

Enable webSocket on the required HTTP profile.

```
> set ns httpProfile http_profile1 -webSocket ENABLED
Done
```

```
> set lb vserver lbvserver1 -httpProfileName profile1
Done
```

SNMP

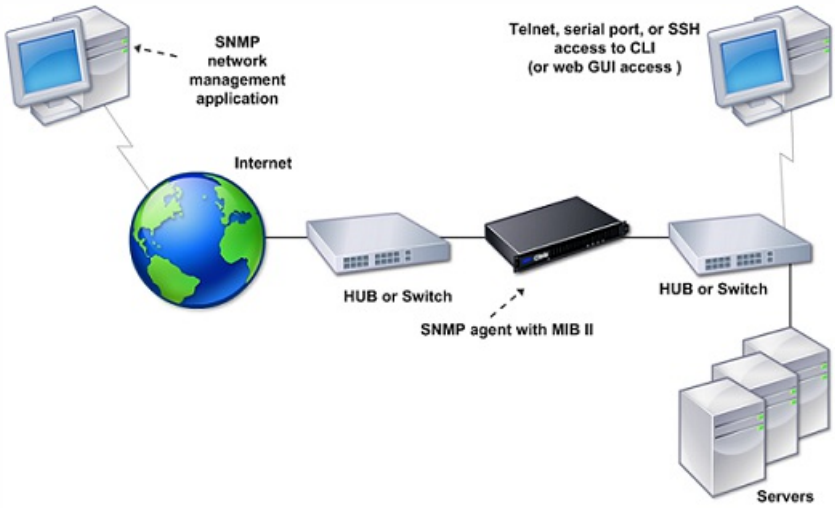
Jun 01, 2015

You can use Simple Network Management Protocol (SNMP) to configure the SNMP agent on the Citrix NetScaler appliance to generate asynchronous events, which are called *traps*. The traps are generated whenever there are abnormal conditions on the NetScaler. The traps are then sent to a remote device called a *trap listener*, which signals the abnormal condition on the NetScaler appliance. Or, you can query the SNMP agent for System-specific information from a remote device called an *SNMP manager*. The agent then searches the management information base (MIB) for the data requested and sends the data to the SNMP manager.

The SNMP agent on the NetScaler can generate traps compliant with SNMPv1, SNMPv2, and SNMPv3. For querying, the SNMP agent supports SNMP version 1 (SNMPv1), SNMP version 2 (SNMPv2), and SNMP version 3 (SNMPv3).

The following figure illustrates a network with a NetScaler that has SNMP enabled and configured. In the figure, each SNMP network management application uses SNMP to communicate with the SNMP agent on the NetScaler. The SNMP agent searches its management information base (MIB) to collect the data requested by the SNMP Manager and provides the information to the application.

Figure 1. *NetScaler Supporting SNMP*



Importing MIB Files to the SNMP Manager and Trap Listener

To monitor a NetScaler appliance, you must download the MIB object definition files. The MIB files include the following:

- MIB-2 groups SYSTEM, IF, ICMP, UDP, and SNMP.
- NetScaler-specific configuration and statistics.

You can obtain the MIB object definition files from the /netscaler/snmp directory or from the Downloads tab of the NetScaler GUI.

If the SNMP management application is other than WhatsUpGold, download the following files to the SNMP management application:

- NS-MIB-smiv1.mib. Used by SNMPv1 managers and trap listeners.

- NS-MIB-smiv2.mib. Used by SNMPv2 and SNMPv3 managers and SNMPv2 trap listeners.

If the SNMP management application is WhatsUpGold, download the following files to the SNMP management application:

- mib.txt
- traps.txt

Configuring the NetScaler to Generate SNMP Traps

Jun 01, 2015

You can configure the NetScaler appliance to generate asynchronous events, which are called *traps*. The traps are generated whenever there are abnormal conditions on the appliance. The traps are sent to a remote device called a *trap listener*. This helps administrators monitor the appliance and respond promptly to any issues.

The NetScaler appliance provides a set of condition entities called *SNMP alarms*. When the condition in any SNMP alarm is met, the appliance generates SNMP trap messages that are sent to the configured trap listeners. For example, when the LOGIN-FAILURE alarm is enabled, a trap message is generated and sent to the trap listener whenever there is a login failure on the appliance.

To configure the NetScaler appliance to generate traps, you need to enable and configure alarms. Then, you specify trap listeners to which the appliance will send the generated trap messages.

This document includes the following details:

- Enabling an SNMP Alarm
- Configuring Alarms
- Configuring SNMPv1 or SNMPv2 Traps
- Enabling Unconditional SNMP Trap Logging

Enabling an SNMP Alarm

Updated: 2014-08-08

The NetScaler appliance generates traps only for SNMP alarms that are enabled. Some alarms are enabled by default, but you can disable them.

When you enable an SNMP alarm, the appliance generates corresponding trap messages when some events occur. Some alarms are enabled by default.

To enable an SNMP alarm by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- enable snmp alarm <trapName>
- show snmp alarm <trapName>

To enable an SNMP alarm by using the configuration utility

1. Navigate to System > SNMP > Alarms, and select the alarm.
2. Click Actions and select Enable.

Configuring Alarms

Updated: 2014-08-06

The NetScaler appliance provides a set of condition entities called *SNMP alarms*. When the condition set for an SNMP alarm is met, the appliance generates SNMP traps messages that are sent to the configured trap listeners. For example, when the LOGIN-FAILURE alarm is enabled, a trap message is generated and sent to the trap listener whenever there is a

login failure on the appliance.

You can assign an SNMP alarm with a severity level. When you do this, the corresponding trap messages are assigned that severity level.

The following are the severity levels, defined on the appliance, in decreasing order of severity.

- Critical
- Major
- Minor
- Warning
- Informational

For example, if you set a warning severity level for the SNMP alarm named LOGIN-FAILURE, the trap messages generated when there is a login failure will be assigned with the warning severity level.

You can also configure an SNMP alarm to log the corresponding trap messages generated whenever the condition on that alarm is met.

To configure an SNMP alarm by using the command line interface

At the command prompt, type the following commands to configure an SNMP alarm and verify the configuration:

- set snmp alarm <trapName> [-thresholdValue <positive_integer> [-normalValue <positive_integer>]] [-time <secs>] [-state (ENABLED | DISABLED)] [-severity <severity>] [-logging (ENABLED | DISABLED)]
- show snmp alarm <trapName>

To configure SNMP alarms by using the configuration utility

Navigate to System > SNMP > Alarms, select an alarm and configure the alarm parameters.

Configuring SNMPv1 or SNMPv2 Traps

Updated: 2014-08-06

After configuring the alarms, you need to specify the trap listener to which the appliance sends the trap messages. Apart from specifying parameters such as IP or IPv6 address and the destination port of the trap listener, you can specify the type of trap (either generic or specific) and the SNMP version.

You can configure a maximum of 20 trap listeners for receiving either generic or specific traps.

You can also configure the appliance to send SNMP trap messages with a source IP address other than the NetScaler IP (NSIP or NSIP6) address to a particular trap listener. For a trap listener that has an IPv4 address, you can set the source IP to either a mapped IP (MIP) address or a subnet IP (SNIP) address configured on the appliance. For a trap listener that has an IPv6 address, you can set the source IP to subnet IPv6 (SNIP6) address configured on the appliance.

You can also configure the appliance to send trap messages to a trap listener on the basis of a severity level. For example, if you set the severity level as Minor for a trap listener, all trap messages of the severity level equal to or greater than Minor (Minor, Major, and Critical) are sent to the trap listener.

If you have defined a community string for the trap listener, you must also specify a community string for each trap that is to be sent to the listener. A trap listener for which a community string has been defined accepts only trap messages that

include a community string matching the community string defined in the trap listener. Other trap messages are dropped.

To add an SNMP trap by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp trap <trapClass> <trapDestination> -version (V1 | V2) -destPort <port> -communityName <string> -srcIP <ip_addr> -severity <severity>
- show snmp trap

Example

```
> add snmp trap specific 10.102.29.3 -version V2 -destPort 80 -communityName com1 -severity Major
```

To configure SNMP Traps by using the configuration utility

Navigate to System > SNMP > Traps, and create the SNMP trap.

Enabling Unconditional SNMP Trap Logging

Updated: 2014-08-08

By default, the NetScaler appliance logs any SNMP trap messages (for SNMP alarms in which logging is enabled) when at least one trap listener is specified on the appliance. However, you can specify that SNMP trap messages be logged even when no trap listeners are configured.

To enable unconditional SNMP trap logging by using the command line interface

At the command prompt, type the following commands to configure unconditional SNMP trap logging and verify the configuration:

- set snmp option -snmpTrapLogging (ENABLED | DISABLED)
- show snmp option

To enable unconditional SNMP trap logging by using the configuration utility

Navigate to System > SNMP, click Change SNMP Options and select SNMP Trap Logging.

Configuring the NetScaler for SNMP v1 and v2 Queries

Jun 01, 2015

You can query the NetScaler SNMP agent for system-specific information from a remote device called *SNMP managers*. The agent then searches the management information base (MIB) for the data requested and sends the data to the SNMP manager.

The following types of SNMP v1 and v2 queries are supported by the SNMP agent:

- GET
- GET NEXT
- ALL
- GET BULK

You can create strings called community strings and associate each of these to query types. You can associate one or more community strings to each query type. Community string are passwords and used to authenticate SNMP queries from SNMP managers.

For example, if you associate two community strings, such as **abc** and **bcd**, to the query type GET NEXT, the SNMP agent on the NetScaler appliance considers only those GET NEXT SNMP query packets that contain **abc** or **bcd** as the community string.

This document includes the following details:

Updated: 2014-08-06

You must configure the NetScaler appliance to allow the appropriate SNMP managers to query it. You must also provide the SNMP manager with the required NetScaler-specific information. You can add up to a maximum of 100 SNMP managers or networks.

For an IPv4 SNMP manager you can specify a host name instead of the manager's IP address. If you do so, you must add a DNS name server that resolves the host name of the SNMP manager to its IP address. You can add up to a maximum of five host-name based SNMP managers.

Note: The appliance does not support use of host names for SNMP managers that have IPv6 addresses. You must specify the IPv6 address.
If you do not configure at least one SNMP manager, the appliance accepts and responds to SNMP queries from all IP addresses on the network. If you configure one or more SNMP managers, the appliance accepts and responds only to SNMP queries from those specific IP addresses.

If you remove an SNMP manager from the configuration, that manager can no longer query the appliance.

To add SNMP managers by specifying IP addresses by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp manager <IPAddress> ... [-netmask <netmask>]
- show snmp manager

```
> add snmp manager 10.102.29.10 10.102.29.15 10.102.29.30
```

To add an SNMP manager by specifying its host name by using the command line interface

Important: If you specify the SNMP manager's host name instead of its IP address, you must configure a DNS name server to resolve the host name to the SNMP manager's IP address. For more information, see "Adding a Name Server."
At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp manager <IPAddress> [-domainResolveRetry **<integer>]**
- show snmp manager

```
> add nameserver 10.103.128.15

> add snmp manager engwiki.eng.example.net –domainResolveRetry 10
```

To add an SNMP manager by using the configuration utility

1. Navigate to System > SNMP > Managers, and create the SNMP manager.
   Important: If you specify the SNMP manager's host name instead of its IPv4 address, you must configure a DNS name server to resolve the host name to the SNMP manager's IP address. For more information, see "Adding a Name Server."
   Note: The appliance does not support host names for SNMP managers that have IPv6 addresses.

Updated: 2014-08-06

You can create strings called community strings and associate them with the following SNMP query types on the appliance:

- GET
- GET NEXT
- ALL
- GET BULK

You can associate one or more community strings to each query types. For example, when you associate two community strings, such as **abc** and **bcd**, to the query type GET NEXT, the SNMP agent on the appliance considers only those GET NEXT SNMP query packets that contain **abc** or **bcd** as the community string.

If you do not associate any community string to a query type then the SNMP agent responds to all SNMP queries of that type.

To specify an SNMP community by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp community <communityName> <permissions>
- show snmp community

```
> add snmp community com all
```

To configure an SNMP community string by using the configuration utility

Navigate to System > SNMP > Community, and create the SNMP community.

Configuring the NetScaler for SNMPv3 Queries

Jun 01, 2015

Simple Network Management Protocol Version 3 (SNMPv3) is based on the basic structure and architecture of SNMPv1 and SNMPv2. However, SNMPv3 enhances the basic architecture to incorporate administration and security capabilities, such as authentication, access control, data integrity check, data origin verification, message timeliness check, and data confidentiality.

To implement message level security and access control, SNMPv3 introduces the user-based security model (USM) and the view-based access control model (VACM).

- **User-Based Security Model.** The user-based security model (USM) provides message-level security. It enables you to configure users and security parameters for the SNMP agent and the SNMP manager. USM offers the following features:
  - **Data integrity:** To protect messages from being modified during transmission through the network.
  - **Data origin verification:** To authenticate the user who sent the message request.
  - **Message timeliness:** To protect against message delays or replays.
  - **Data confidentiality:** To protect the content of messages from being disclosed to unauthorized entities or individuals.
- **View-Based Access Control Model.** The view-based access control model (VACM) enables you to configure access rights to a specific subtree of the MIB based on various parameters, such as security level, security model, user name, and view type. It enables you to configure agents to provide different levels of access to the MIB to different managers.

The Citrix NetScaler supports the following entities that enable you to implement the security features of SNMPv3:

- SNMP Engines
- SNMP Views
- SNMP Groups
- SNMP Users

These entities function together to implement the SNMPv3 security features. Views are created to allow access to subtrees of the MIB. Then, groups are created with the required security level and access to the defined views. Finally, users are created and assigned to the groups.

Note: The view, group, and user configuration are synchronized and propagated to the secondary node in a high availability (HA) pair. However, the engine ID is neither propagated nor synchronized as it is unique to each NetScaler appliance.
To implement message authentication and access control, you need to:

- Set the Engine ID
- Configure Views
- Configure Groups
- Configure Users

This document includes the following details:

- Setting the Engine ID
- Configuring a View
- Configuring a Group
- Configuring a User

Updated: 2014-08-06

SNMP engines are service providers that reside in the SNMP agent. They provide services such as sending, receiving, and authenticating messages. SNMP engines are uniquely identified using engine IDs.

The NetScaler appliance has a unique engineID based on the MAC address of one of its interfaces. It is not necessary to override the engineID. However, if you want to change the engine ID, you can reset it.

To set the engine ID by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- set snmp engineId <engineID>
- show snmp engineId

```
> set snmp engineId 8000173f0300c095f80c68
```

To set the engine ID by using configuration utility

Navigate to System > SNMP > Users, click Configure Engine ID and type an engine ID.

Updated: 2014-08-06

SNMP views restrict user access to specific portions of the MIB. SNMP views are used to implement access control.

To add an SNMP view by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp view <name> <subtree> -type (included | excluded)
- show snmp view <name>

```
> add snmp view View1 -type included
```

To configure an SNMP view by using the configuration utility

Navigate to System > SNMP > Views, and create the SNMP view.

Updated: 2014-08-06

SNMP groups are logical aggregations of SNMP users. They are used to implement access control and to define the security levels. You can configure an SNMP group to set access rights for users assigned to that group, thereby restricting the users to specific views.

You need to configure an SNMP group to set access rights for users assigned to that group.

To add an SNMP group by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp group <name> <securityLevel> -readViewName <string>
- show snmp group <name> <securityLevel>


```
> add snmp group edocs_group2 authPriv -readViewName edocs_read_view
```

To configure an SNMP group by using the configuration utility

Navigate to System > SNMP > Groups, and create the SNMP group.

Updated: 2014-08-06

SNMP users are the SNMP managers that the agents allow to access the MIBs. Each SNMP user is assigned to an SNMP group.

You need to configure users at the agent and assign each user to a group.

To configure a user by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add snmp user <name> -group <string> [-authType (MD5 | SHA) {-authPasswd } [-privType (DES | AES) {-privPasswd }]]
- show snmp user <name>

```
> add snmp user edocs_user -group edocs_group
```

To configure an SNMP user by using the configuration utility

Navigate to System > SNMP > Users, and create the SNMP user.

Configuring SNMP Alarms for Rate Limiting

Jun 01, 2015

Citrix NetScaler appliances such as the NetScaler MPX 10500, 12500, and 15500 are rate limited. The maximum throughput (Mbps) and packets per second (PPS) are determined by the license purchased for the appliance. For rate-limited platforms, you can configure SNMP traps to send notifications when throughput and PPS approach their limits and when they return to normal.

Throughput and PPS are monitored every seven seconds. You can configure traps with high-threshold and normal-threshold values, which are expressed as a percentage of the licensed limits. The appliance then generates a trap when throughput or PPS exceeds the high threshold, and a second trap when the monitored parameter falls to the normal threshold. In addition to sending the traps to the configured destination device, the NetScaler logs the events associated with the traps in the /var/log/ns.log file as EVENT ALERTSTARTED and EVENT ALERTENDED.

Exceeding the throughput limit can result in packet loss. You can configure SNMP alarms to report packet loss.

For more information about SNMP alarms and traps, see "Configuring the NetScaler to generate SNMP v1 and v2 Traps."

This document includes the following details:

- Configuring an SNMP Alarm for Throughput or PPS
- Configuring SNMP Alarm for Dropped Packets

Updated: 2014-08-12

To monitor both throughput and PPS, you must configure separate alarms.

To configure an SNMP alarm for the throughput rate by using the command line interface

At the command prompt, type the following commands to configure the SNMP alarm and verify the configuration:

- set snmp alarm PF-RL-RATE-THRESHOLD [-thresholdValue <positive_integer> [-normalValue <positive_integer>]] [-state (**ENABLED** | **DISABLED**)] [-severity <severity>] [-logging (**ENABLED** | **DISABLED**)]
- show snmp alarm PF-RL-RATE-THRESHOLD

```
> set snmp alarm PF-RL-RATE-THRESHOLD -thresholdValue 70 -normalValue 50
```

To configure an SNMP alarm for PPS by using the command line interface

At the command prompt, type the following commands to configure the SNMP alarm for PPS and verify the configuration:

- set snmp alarm PF-RL-PPS-THRESHOLD [-thresholdValue <positive_integer> [-normalValue <positive_integer>]] [-state (**ENABLED** | **DISABLED**)] [-severity <severity>] [-logging (**ENABLED** | **DISABLED**)]
- show snmp alarm PF-RL-PPS-THRESHOLD

```
> set snmp alarm PF-RL-PPS-THRESHOLD -thresholdValue 70 -normalValue 50
```

To configure an SNMP alarm for throughput or PPS by using the configuration

utility

1. Navigate to System > SNMP > Alarms, and select PF-RL-RATE-THRESHOLD (for throughput rate) or PF-RL-PPS-THRESHOLD (for packets per second).
2. Set the alarm parameters and enable the selected SNMP alarm.

Updated: 2014-08-12

You can configure an alarm for packets dropped as a result of exceeding the throughput limit and an alarm for packets dropped as a result of exceeding the PPS limit.

To configure an SNMP alarm for packets dropped because of excessive throughput, by using the command line interface

At the command prompt, type:
set snmp alarm PF-RL-RATE-PKTS-DROPPED [-state (**ENABLED** | **DISABLED**)] [-severity <severity>] [-logging (**ENABLED** | **DISABLED**)]

To configure an SNMP alarm for packets dropped because of excessive PPS, by using the command line interface

At the command prompt, type:
set snmp alarm PF-RL-PPS-PKTS-DROPPED [-state (**ENABLED** | **DISABLED**)] [-severity <severity>] [-logging (**ENABLED** | **DISABLED**)]

To configure an SNMP alarm for dropped packets by using the configuration utility

1. Navigate to System > SNMP > Alarms, and select PF-RL-RATE-PKTS-DROPPED (for packets dropped because of excessive throughput) or PF-RL-PPS-PKTS-DROPPED (for packets dropped because of excessive PPS).
2. Set the alarm parameters and enable the selected SNMP alarm.

Audit Logging

Jun 03, 2015

Auditing is a methodical examination or review of a condition or situation. The Audit Logging feature enables you to log the NetScaler states and status information collected by various modules in the kernel and in the user-level daemons. For audit logging, you can use the SYSLOG protocol, the native NSLOG protocol, or both.

SYSLOG is a standard protocol for logging. It has two components: the SYSLOG auditing module, which runs on the NetScaler appliance, and the SYSLOG server, which can run on the underlying FreeBSD operating system (OS) of the NetScaler appliance or on a remote system. SYSLOG uses user data protocol (UDP) for data transfer.

Similarly, the native NSLOG protocol has two components— the NSLOG auditing module, which runs on the NetScaler appliance, and the NSLOG server, which can run on the underlying FreeBSD OS of the NetScaler appliance or on a remote system. NSLOG uses transmission control protocol (TCP) for data transfer.

When you run a SYSLOG or NSLOG server, it connects to the NetScaler appliance. The NetScaler appliance then starts sending all the log information to the SYSLOG or NSLOG server, and the server can filter the log entries before storing them in a log file. An NSLOG or SYSLOG server can receive log information from more than one NetScaler appliance, and a NetScaler appliance can send log information to more than one SYSLOG server or NSLOG server.

The log information that a SYSLOG or NSLOG server collects from a NetScaler appliance is stored in a log file in the form of messages. These messages typically contain the following information:

- The IP address of a NetScaler appliance that generated the log message
- A time stamp
- The message type
- The predefined log levels (Critical, Error, Notice, Warning, Informational, Debug, Alert, and Emergency)
- The message information

To configure audit logging, you first configure the audit modules on the NetScaler appliance. That involves creating audit policies and specifying the NSLOG server or SYSLOG server information. You then install and configure the SYSLOG or the NSLOG server on the underlying FreeBSD OS of the NetScaler appliance or on a remote system.

Note: Because SYSLOG is an industry standard for logging program messages, and various vendors provide support, this documentation does not include SYSLOG server configuration information.
The NSLOG server has its own configuration file (auditlog.conf). You can customize logging on the NSLOG server system by making additional modifications to the configuration file (auditlog.conf).

Configuring the NetScaler Appliance for Audit Logging

Jun 01, 2015

On the NetScaler appliance, you configure SYSLOG and/or NSLOG policies. Each policy includes a rule, which is an expression identifying the messages to be logged, and a SYSLOG or NSLOG (depending on the type of policy) action. The action specifies the server to which to send the log message, the level of the messages to be logged, and the data format of the logged messages. You can bind the policies globally or to individual virtual servers.

The appliance logs the following information related to TCP connections:

- Source port
- Destination port
- Source IP
- Destination IP
- Number of bytes transmitted and received
- Time period for which the connection is open

Note:

- You can enable TCP logging on individual load balancing virtual servers. You must bind the audit log policy to a specific load balancing virtual server that you want to log.
- When using the NetScaler as the audit log server, by default, the ns.log file is rotated (new file is created) when the file size reaches 100K and the last 25 copies of the ns.log are archived and compressed with gzip. To accommodate more archived files after 25 files, the oldest archive is deleted. You can modify the 100K limit or the 25 file limit by updating the following entry in the /etc/newsyslog.conf file:
  `/var/log/ns.log 600 25 100 * Z`
  where, 25 is the number of archived files to be maintained and 100K is the size of the ns.log file after which the file will be archived.

This document includes the following details:

- Configuring Audit Servers
- Configuring Audit Policies
- Binding the Audit Policies Globally
- Configuring Policy-Based Logging

Updated: 2015-06-03

You can configure audit server actions for different servers and for different log levels.

To configure a SYSLOG server action by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add audit syslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> [-dateFormat (MMDDYYYY | DDMMYYYY)]
- show audit syslogAction [<name>]

```
> add audit syslogaction audit-action1 10.102.1.1 -loglevel INFORMATIONAL -dateformat MMDDYYYY
```

To configure an NSLOG server action by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add audit nslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> [-dateFormat (MMDDYYYY | DDMMYYYY)]
- show audit nslogAction [<name>]

```
> add audit nslogAction nslog-action1 10.102.1.3 -serverport 520 -loglevel INFORMATIONAL -dateFormat MMDDYYYY
```

To configure an auditing server action by using the configuration utility

Navigate to System > Auditing > Syslog or Nslog, click Servers tab and create the auditing server.

Updated: 2015-04-29

Configure SYSLOG policies to log messages to a SYSLOG server, and/or NSLOG policy to log messages to an NSLOG server. Each policy includes a rule identifying the messages to be logged, and a SYSLOG or NS LOG (depending on the type of policy) action.

To configure a SYSLOG policy by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add audit syslogPolicy <name> <rule> <action>
- show audit syslogPolicy [<name>]

```
> add audit syslogpolicy syslog-pol1 ns_true audit-action1
```

To configure an NSLOG policy by using the command line interface

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add audit nslogPolicy <name> <rule> <action>
- show audit nslogPolicy [<name>]

```
> add audit nslogPolicy nslog-pol1 ns_true nslog-action1
```

To configure an audit server policy by using the configuration utility

Navigate to System > Auditing > Syslog or Nslog, click Policies tab and create the auditing policy.

Updated: 2015-06-02

You must globally bind the audit log policies to enable logging of all NetScaler system events. By defining the priority level, you can set the evaluation order of the audit server logging. Priority 0 is the highest and is evaluated first. The higher the priority number, the lower is the priority of evaluation.

To configure a SYSLOG policy by using the command line interface

At the command prompt, type:

- bind system global [<policyName> [-priority <positive_integer>]]
- show system global

```
> bind system global nslog-pol1 -priority 20
```

To globally bind the audit policy by using the configuration utility

1. Navigate to System > Auditing > Syslog or Nslog.
2. On Policies tab, click Action, and select Global Bindings to bind the audit global policies.

Updated: 2014-08-07

You can configure policy-based logging for rewrite and responder policies. Audit messages are then logged in a defined format when the rule in a policy evaluates to TRUE. To configure policy-based logging, you configure an audit-message action that uses default syntax expressions to specify the format of the audit messages, and associate the action with a policy. The policy can be bound either globally or to a load balancing or content switching virtual server. You can use audit-message actions to log messages at various log levels, either in syslog format only or in both syslog and newnslog formats.

Pre Requisites

- User Configurable Log Messages (userDefinedAuditlog) option is enabled for when configuring the audit action server to which you want to send the logs in a defined format. For more information about enabling policy-based logging on an audit action server, see "Binding the Audit Policies Globally."
- The related audit policy is bound to system global. For more information about binding audit policies to system global, see "Binding the Audit Policies Globally."

Configuring an Audit Message Action

You can configure audit message actions to log messages at various log levels, either in syslog format only or in both syslog and newnslog formats. Audit-message actions use expressions to specify the format of the audit messages.

At the command prompt, type:

add audit messageaction <name> <logLevel> <stringBuilderExpr> [-logtoNewnslog (YES | NO)] [-bypassSafetyCheck (YES | NO)]

Example

```
> add audit messageaction log-act1 CRITICAL '"Client:"+CLIENT.IP.SRC+" accessed "+HTTP.REQ.URL' -bypassSafetyCheck YES
```

Navigate to System > Auditing > Message Actions, and create the audit message action.

Binding Audit Message Action to a Policy

After you have created an audit message action, you must bind it to a rewrite or responder policy. For more information about binding log message actions to a rewrite or responder policy, see "Rewrite" or "Responder".

Installing and Configuring the NSLOG Server

Jul 23, 2015

During installation, the NSLOG server executable file (auditserver) is installed along with other files. The auditserver executable file includes options for performing several actions on the NSLOG server, including running and stopping the NSLOG server. In addition, you use the auditserver executable to configure the NSLOG server with the IP addresses of the NetScaler appliances from which the NSLOG server will start collecting logs. Configuration settings are applied in the NSLOG server configuration file (auditlog.conf).

Then, you start the NSLOG server by executing the auditserver executable. The NSLOG server configuration is based on the settings in the configuration file. You can further customize logging on the NSLOG server system by making additional modifications to the NSLOG server configuration file (auditlog.conf).

Attention: The version of the NSLOG server package must be the same as that of the NetScaler. For example, if the version of the NetScaler is 10.1 Build 125.9, the NSLOG server must also be of the same version.
The following table lists the operating systems on which the NSLOG server is supported.

Table 1. Supported Platforms for the NSLOG Server

Operating system	Software requirements	Remarks
Windows	• Windows XP Professional • Windows Server 2003 • Windows 2000/NT • Windows Server 2008 • Windows Server 2008 R2	
Linux	• RedHat Linux 4 or later • SUSE Linux Enterprise 9.3 or later	
FreeBSD	FreeBSD 6.3 or later	For NetScaler 10.5, use only FreeBSD 8.4.
Mac OS	Mac OS 8.6 or later	Not supported on NetScaler 10.1 and later releases.

The minimum hardware specifications for the platform running the NSLOG server are as follows:

- Processor- Intel x86 ~501 megahertz (MHz)
- RAM - 512 megabytes (MB)
- Controller - SCSI

This document includes the following details:

- Installing NSLOG Server on the Linux Operating System
- Installing NSLOG Server on the FreeBSD Operating System
- Installing NSLOG Server Files on the Windows Operating System
- NSLOG Server Command Options
- Adding the NetScaler Appliance IP Addresses on the NSLOG Server
- Verifying the NSLOG Server Configuration File

Updated: 2013-06-20

Log on to the Linux system as an administrator. Use the following procedure to install the NSLOG server executable files on the system.

To install the NSLOG server package on a Linux operating system

1. At a Linux command prompt, type the following command to copy the NSauditserver.rpm file to a temporary directory:
   cp <path_to_cd>/Utilities/auditserver/Linux/NSauditserver.rpm /tmp
2. Type the following command to install the NSauditserver.rpm file:
   rpm -i NSauditserver.rpm

   This command extracts the files and installs them in the following directories:

   - /usr/local/netscaler/etc
   - /usr/local/netscaler/bin
   - /usr/local/netscaler/samples

To uninstall the NSLOG server package on a Linux operating system

1. At a command prompt, type the following command to uninstall the audit server logging feature:
   rpm -e NSauditserver
2. For more information about the NSauditserver RPM file, use the following command:
   rpm -qpi *.rpm
3. To view the installed audit server files use the following command:
   rpm -qpl *.rpm

   *.rpm: Specifies the file name.

Before you can install the NSLOG server, you have to copy the NSLOG package from the NetScaler product CD or download it from www.citrix.com. The NSLOG package has the following name format:

AuditServer_<release number>-<build number>.zip

For example: AuditServer_10.1-132.8.zip

This package contains files for all supported platforms: Linux, Windows, and FreeBSD. On a FreeBSD operating system, install the NSLOG package that has the following name format:

audserver_bsd-<release number>-<build number>.tgz

For example: audserver_bsd-10.1-132.8.tgz

To download NSLOG package from www.citrix.com

1. In a web browser, go to www.citrix.com.
2. In the menu bar, click **Log In**.
3. Enter your login credentials, and then click **Log In**.
4. In the menu bar, click **Downloads**.
5. From the **Select a product** list, select **NetScaler ADC**.
6. On the **NetScaler ADC** page, select the release for which you want to download the NSLOG package (for example, Release 10.1), and then select **Firmware**.
7. Under **Firmware**, select the NetScaler firmware for the build number for which you want to download the NSLOG package.
8. On the page that appears, scroll down, select **Audit Servers**, and click **Download File** next to the package that you want to download.

To install the NSLOG server package on a FreeBSD operating system

1. On the system to which you have downloaded the NSLOG package **AuditServer_<release number>-<build number>.zip** (for example, **AuditServer_9.3-51.5.zip**), extract the FreeBSD NSLOG server package **audserver_bsd-<release number>-<build number>.tgz** (for example, **audserver_bsd-9.3-51.5.tgz**) from the package.
2. Copy the FreeBSD NSLOG server package **audserver_bsd-<release number>-<build number>.tgz** (for example, **audserver_bsd-9.3-51.5.tgz**) to a directory on a system running FreeBSD OS.
3. At a command prompt for the directory into which the FreeBSD NSLOG server package was copied, run the following command to install the package:
   pkg_add audserver_bsd-<release number>-<build number>.tgz

   Example
   pkg_add audserver_bsd-9.3-51.5.tgz
   The following directories are extracted:

   - <root directory extracted from the FreeBSD NSLOG server package tgz file>\netscaler\bin (for example, /var/auditserver/netscaler/bin)
   - <root directory extracted from the FreeBSD NSLOG server package tgz file>\netscaler\etc (for example, /var/auditserver/netscaler/etc)
   - <root directory extracted from the FreeBSD NSLOG server package tgz file>\netscaler\samples (for example, /var/auditserver/samples)
4. At a command prompt, type the following command to verify that the package is installed:
   pkg_info | grep NSaudserver

To uninstall the NSLOG server package on a FreeBSD operating system

At a command prompt, type:

pkg_delete NSaudserver

Updated: 2013-06-20

Before you can install the NSLOG server, you have to copy the NSLOG package from the NetScaler product CD or download it from www.citrix.com. The NSLOG package has the following name format AuditServer _<release number>-<build number>.zip (for example, AuditServer_9.3-51.5.zip). This package contains NSLOG installation packages for all supported platforms.

To download NSLOG package from www.Citrix.com

1. In a web browser, go to www.citrix.com.
2. In the menu bar, click Log In.
3. Enter your login credentials, and then click Log In.
4. In the menu bar, click Downloads.
5. Search to find the page that provides the appropriate release number and build.
6. On that page, under Audit Servers, click Download to download the NSLOG package, having the format AuditServer_<release number>-<build number>.zip , to your local system (for example, AuditServer_9.3-51.5.zip).

To install NSLOG server on a Windows operating system

1. On the system, where you have downloaded the NSLOG package AuditServer_<release number>-<build number>.zip (for example, AuditServer_9.3-51.5.zip), extract audserver_win-<release number>-<build number>.zip (for example, audserver_win-9.3-51.5.zip) from the package.
2. Copy the extracted file audserver_<release number>-<build number>.zip (for example, audserver_win-9.3-51.5.zip) to a Windows system on which you want to install the NSLOG server.
3. Unzip the audserver_<release number>-<build number>.zip file (for example, audserver_win-9.3-51.5.zip).
4. The following directories are extracted:
   1. <root directory extracted from the Windows NSLOG server package zip file>\bin (for example, C:\audserver_win-9.3-51.5\bin)
   2. <root directory extracted from the Windows NSLOG server package zip file>\etc (for example, C:\audserver_win-9.3-51.5\ etc)
   3. < root directory extracted from the Windows NSLOG server package zip file >\samples (for example, C:\audserver_win-9.3-51.5\ samples)
5. At a command prompt, run the following command from the <root directory extracted from the Windows NSLOG server package zip file>\bin path:
   audserver -install -f <directorypath>\auditlog.conf

   <directorypath>: Specifies the path to the configuration file (auditlog.conf). By default, log.conf is under <root directory extracted from Windows NSLOG server package zip file>\samples directory. But you can copy auditlog.conf to your desired directory.

To uninstall the NSLOG server on a Windows operating system

At a command prompt, run the following from the <root directory extracted from Windows NSLOG server package zip file>\bin path:

audserver -remove

Updated: 2013-06-20

The following table describes the commands that you can use to configure audit server options.

Table 2. Audit Server Options

Audit server commands	Specifies
audserver -help	The available Audit Server options.
audserver -addns -f <path to configuration file>	The system that gathers the log transaction data. You are prompted to enter the IP address of the NetScaler appliance. Enter the valid user name and password.
audserver -verify -f <path to configuration file>	Check for syntax or semantic errors in the configuration file (for example, auditlog.conf).
audserver -start -f <path to configuration file>	Start audit server logging based on the settings in the configuration file (auditlog.conf). Linux only: To start the audit server as a background process, type the ampersand sign (&) at the end of the command.
audserver -stop (Linux only)	Stops audit server logging when audit server is started as a background process. Alternatively, use the Ctrl+C key to stop audit server logging.
audserver -install -f <path to configuration file> (Windows only)	Installs the audit server logging client as a service on Windows.
audserver -startservice (Windows Only)	Start the audit server logging service, when you enter this command at a command prompt. You can also start audit server logging from Start > Control Panel > Services. Note: Audit server logging starts by using the configuration settings in the configuration file, for example, auditlog.conf file specified in the audit server install option.
audserver -stopservice (Windows Only)	Stop audit server logging.
audserver -remove	Removes the audit server logging service from the registry.

Audit server commands	Specifies
Run the audserver command	from the directory in which the audit server executable is present:

- On Windows: \ns\bin
- On Solaris and Linux: \usr\local\netscaler\bin

The audit server configuration files are present in the following directories:

- On Windows: \ns\etc
- On Linux: \usr\local\netscaler\etc

The audit server executable is started as ./auditserver in Linux and FreeBSD.

In the configuration file (auditlog.conf), add the IP addresses of the NetScaler appliances whose events must be logged.

To add the IP addresses of the NetScaler appliance

At a command prompt, type the following command:

audserver -addns -f <directorypath>\auditlog.conf

<directorypath>: Specifies the path to the configuration file (auditlog.conf).

You are prompted to enter the information for the following parameters:

NSIP: Specifies the IP address of the NetScaler appliance, for example, 10.102.29.1.

Userid: Specifies the user name, for example, nsroot.

Password: Specifies the password, for example, nsroot.

If you add multiple NetScaler IP addresses (NSIP), and later you do not want to log all of the NetScaler appliance event details, you can delete the NSIPs manually by removing the NSIP statement at the end of the auditlog.conf file. For a high availability (HA) setup, you must add both primary and secondary NetScaler IP addresses to auditlog.conf by using the audserver command. Before adding the IP address, make sure the user name and password exist on the system.

Check the configuration file (audit log.conf) for syntax correctness to enable logging to start and function correctly.

To verify configuration, at a command prompt, type the following command:

audserver -verify -f <directorypath>\auditlog.conf

<directorypath>: Specifies the path to the configuration file (audit log.conf).

Running the NSLOG Server

Jun 20, 2013

Type the following command at a command prompt:

audserver -start -f <directorypath>\auditlog.conf
<directorypath>: Specifies the path to the configuration file (audit log.conf).

Type the following command:

audserver -stop

Type the following command:

audserver -stopservice

Customizing Logging on the NSLOG Server

Jun 01, 2015

You can customize logging on the NSLOG server by making additional modifications to the NSLOG server configuration file (log.conf). Use a text editor to modify the log.conf configuration file on the server system.

To customize logging, use the configuration file to define filters and log properties.

- **Log filters.** Filter log information from a NetScaler appliance or a set of NetScaler appliances.
- **Log properties.** Each filter has an associated set of log properties. Log properties define how to store the filtered log information.

This document includes the following details:

Updated: 2013-11-14

You can use the default filter definition located in the configuration file (audit log.conf), or you can modify the filter or create a new filter. You can create more than one log filter.

Note: For consolidated logging, if a log transaction occurs for which there is no filter definition, the default filter is used (if it is enabled.) The only way you can configure consolidated logging of all the NetScaler appliances is by defining the default filter.

To create a filter

At the command prompt, type the following command in the configuration file (auditlog.conf) :

filter <filterName> [IP <ip>] [NETMASK <mask>] [ON | OFF]

<filterName>: Specify the name of the filter (maximum of 64 alphanumeric characters).

<ip>: Specify the IP addresses.

<mask>: Specify the subnet mask to be used on a subnet.

Specify ON to enable the filter to log transactions, or specify OFF to disable the filter. If no argument is specified, the filter is ON

filter F1 IP 192.168.100.151 ON

To apply the filter F2 to IP addresses 192.250.100.1 to 192.250.100.254:

filter F2 IP 192.250.100.0 NETMASK 255.255.255.0 ON

filterName is a required parameter if you are defining a filter with other optional parameters, such as IP address, or the combination of IP address and Netmask.

Updated: 2013-11-13

Log properties associated with the filter are applied to all the log entries present in the filter. The log property definition starts with the key word BEGIN and ends with END as illustrated in the following example:

```
BEGIN <filtername>
 logFilenameFormat ...
 logDirectory ...
 logInterval ...
 logFileSizeLimit ....
END
```

Entries in the definition can include the following:

- **LogFilenameFormat** specifies the file name format of the log file. The name of the file can be of the following types:
  - Static: A constant string that specifies the absolute path and the file name.
  - Dynamic: An expression that includes the following format specifiers:
    - Date (%{format}t)
    - % creates file name with NSIP

  Example

  ```
  LogFileNameFormat Ex%{%m%d%y}t.log
  ```

  This creates the first file name as Exmmddyy.log. New files are named: Exmmddyy.log.0, Exmmddyy.log.1, and so on. In the following example, the new files are crated when the file size reaches 100MB.

  Example

  ```
  LogInterval size
  LogFileSize 100
  LogFileNameFormat Ex%{%m%d%y}t
  ```

  Caution: The date format %t specified in the LogFilenameFormat parameter overrides the log interval property for that filter. To prevent a new file being created every day instead of when the specified log file size is reached, do not use %t in the LogFilenameFormat parameter.
- **logDirectory** specifies the directory name format of the log file. The name of the file can be either of the following:
  - Static: Is a constant string that specifies the absolute path and file name.
  - Dynamic: Is an expression containing the following format specifiers:
    - Date (%{format}t)
    - % creates directory with NSIP

  The directory separator depends on the operating system. In Windows, use the directory separator \.

  Example:

  ```
  LogDirectory dir1\dir2\dir3
  ```

  In the other operating systems (Linux, FreeBsd, etc.), use the directory separator /.

- **LogInterval** specifies the interval at which new log files are created. Use one of the following values:
  - Hourly: A file is created every hour. Default value.
  - Daily: A file is created very day at midnight.
  - Weekly: A file is created every Sunday at midnight.

- Monthly : A file is created on the first day of the month at midnight.
- None: A file is created only once, when audit server logging starts.
- Size: A file is created only when the log file size limit is reached.

Example

```
LogInterval Hourly
```

- **LogFileSizeLimit** specifies the maximum size (in MB) of the log file. A new file is created when the limit is reached.

  Note that you can override the loginterval property by assigning size as its value.

  The default LogFileSizeLimit is 10 MB.

  Example

  ```
  LogFileSizeLimit 35
  ```

Default Settings for the Log Properties

Mar 28, 2012

The following is an example of the default filter with default settings for the log properties:

```
begin default
 logInterval Hourly
 logFileSizeLimit 10
 logFilenameFormat auditlog%{%y%m%d}t.log
 end default
```

Following are two examples of defining the default filters:

Example 1

```
Filter f1 IP 192.168.10.1
```

This creates a log file for NSI 192.168.10.1 with the default values of the log in effect.

Example 2

```
Filter f1 IP 192.168.10.1
begin f1
 logFilenameFormat logfiles.log
end f1
```

This creates a log file for NSIP 192.168.10.1. Since the log file name format is specified, the default values of the other log properties are in effect.

Sample Configuration File (audit.conf)

Mar 28, 2012

Following is a sample configuration file:

```
################################
# This is the Auditserver configuration file
# Only the default filter is active
# Remove leading # to activate other filters
################################
MYIP <NSAuditserverIP>
MYPORT 3023
# Filter filter_nsip  IP <Specify the NetScaler IP address to filter on > ON
# begin filter_nsip
#  logInterval   Hourly
#  logFileSizeLimit  10
#  logDirectory   logdir\%A\
#  logFilenameFormat  nsip%{%d%m%Y}t.log
# end filter_nsip
Filter default
begin default
 logInterval   Hourly
 logFileSizeLimit  10
 logFilenameFormat  auditlog%{%y%m%d}t.log
end default
```

Web Server Logging

Jun 25, 2014

You can use the Web server logging feature to send logs of HTTP and HTTPS requests to a client system for storage and retrieval. This feature has two components:

- The Web log server, which runs on the NetScaler.
- The NetScaler Web Logging (NSWL) client, which runs on the client system.

When you run the NetScaler Web Logging (NSWL) client:

1. It connects to the NetScaler.
2. The NetScaler buffers the HTTP and HTTPS request log entries before sending them to the client.
3. The client can filter the entries before storing them.

To configure Web server logging, you first enable the Web logging feature on the NetScaler and configure the size of the buffer for temporarily storing the log entries. Then, you install NSWL on the client system. You then add the NetScaler IP address (NSIP) to the NSWL configuration file. You are now ready to start the NSWL client to begin logging. You can customize Web server logging by making additional modifications to the NSWL configuration file (log.conf).

Configuring the NetScaler for Web Server Logging

Aug 07, 2014

To configure the NetScaler for web server logging you are required to only enable the Web Server Logging feature. Optionally, you can perform the following configurations:

- Modify the size of the buffer (default size is 16 MB) that stores the logged information before it is sent to the NetScaler Web Logging (NSWL) client.
- Specify the custom HTTP headers that you want to export to the NSWL client. You can configure a maximum of two HTTP request and two HTTP response header names.

At the command prompt, perform the following operations:

- Enable the web server logging feature.
  enable ns feature WL
- [Optional] Modify the buffer size for storing the logged information.
  set ns weblogparam -bufferSizeMB <size>

  Note: To activate your modification, you must disable and then re-enable the Web server logging feature.
- [Optional] Specify the custom HTTP header names that you want to export.
  set ns weblogparam [-customReqHdrs <string> ...] [-customRspHdrs <string> ...]

  Example
  > set ns weblogparam -customReqHdrs Accept-Encoding X-Forwarded -customRspHdrs Content-Encoding ETag

Navigate to System > Settings and perform the following operations:

- To enable the web server logging feature, click Change Advanced Features and select Web Logging.
- To modify the buffer size, click Change Global System Settings and under Web Logging, enter the buffer size.
- To specify the custom HTTP headers to be exported, click Change Global System Settings and under Web Logging, specify the header values.

Installing the NetScaler Web Logging (NSWL) Client

Jun 01, 2015

During installation, the NSWL client executable file (nswl) is installed along with other files. The nswl executable file provides a list of options that you can use. For details, see Configuring the NSWL Client.

Attention: The version of the NSWL client must be the same as that of the NetScaler. For example, if the version of the NetScaler is 10.1 Build 125.9, the NSWL client must also be of the same version.
The following table lists the operating systems on which the NSWL client can be installed.

Table 1. Supported Platforms for the NSWL Client with hardware requirements

Operating system	Version	Hardware requirements	Remarks
Windows	• Windows XP Professional • Windows Server 2003 • Windows 2000/NT • Windows Server 2008 • Windows Server 2008 R2	Processor - Intel x86 ~501 MHz RAM - 512 MB Controller - SCSI	
Mac OS	Mac OS 8.6 or later	-	Not supported on NetScaler 10.1 and later releases.
Linux	• RedHat Linux 4 or later • SUSE Linux Enterprise 9.3 or later	Processor - Intel x86 ~501 MHz RAM - 512 MB Controller - SCSI	
Solaris	Solaris Sun OS 5.6 or later	Processor - UltraSPARC-IIi 400 MHz RAM - 512 MB Controller - SCSI	Not supported on NetScaler 10.5 and later releases.
FreeBSD	FreeBSD 6.3 or later	Processor - Intel x86 ~501 MHz RAM - 512 MB Controller - SCSI	For NetScaler 10.5, use only FreeBSD 8.4.

Operating system	Version	Hardware requirements	Remarks
AIX	AIX 6.1	-	Not supported on NetScaler 10.5 and later releases.

If the NSWL client system cannot process the log transaction because of a CPU limitation, the Web log buffer overruns and the logging process reinitiates.

Caution: Reinitiation of logging can result in loss of log transactions.
To temporarily solve a NSWL client system bottleneck caused by a CPU limitation, you can tune the Web server logging buffer size on the NetScaler appliance. To solve the problem, you need a client system that can handle the site's throughput.

This document includes the following details:

- Downloading the NSWL Client
- Installing the NSWL Client on a Solaris System
- Installing the NSWL Client on a Linux System
- Installing the NSWL Client on a FreeBSD System
- Installing the NSWL Client on a Mac System
- Installing the NSWL Client on a Windows System
- Installing the NSWL Client on a AIX System

Updated: 2014-06-25

You can obtain the NSWL client package from either the NetScaler product CD or the Citrix downloads site. Within the package there are separate installation packages for each supported platforms.

To download the NSWL client package from the Citrix site

1. Open the URL: https://www.citrix.com/downloads.html.
2. Log in to the site using your credentials.
3. Open the page for the required release number and build.
4. In the page, under Weblog Clients, click Download. The package has the name format as follows: Weblog-<release number>-<build number>.zip.

Updated: 2014-06-25

To install the NSWL client, perform the following operations on the system where you downloaded the package.

1. Extract the nswl_solaris-<release number>-<build number>.tar file from the package.
2. Copy the extracted file to a Solaris system on which you want to install the NSWL client.
3. Extract the files from the tar file with the following command:
   `tar xvf nswl_solaris-9.3-51.5.tar`
   A directory NSweblog is created in the temporary directory, and the files are extracted to the NSweblog directory.
4. Install the package with the following command:
   `pkgadd -d`

The list of available packages appears. In the following example, one NSweblog package is shown:

```
1 NSweblog NetScaler Weblogging (SunOS,sparc) 7.0
```

5. You are prompted to select the packages. Select the package number of the NSweblog to be installed.
   After you select the package number and press Enter, the files are extracted and installed in the following directories:
   - /usr/local/netscaler/etc
   - /usr/local/netscaler/bin
   - /usr/local/netscaler/samples
6. To check whether the NSWL package is installed, execute the following command:
   `pkginfo | grep NSweblog`
   Note: To uninstall the NSWL package, execute the following command:
   `pkgrm NSweblog`

Updated: 2014-06-25

To install the NSWL client, perform the following operations on the system where you downloaded the package.

1. Extract the nswl_linux-<release number>-<build number>.rpm file from the package.
2. Copy the extracted file to a system, running Linux OS, on which you want to install the NSWL client.
3. To install the NSWL package, execute the following command:
   `rpm -i nswl_linux-9.3-51.5.rpm`
   This command extracts the files and installs them in the following directories.
   - /usr/local/netscaler/etc
   - /usr/local/netscaler/bin
   - /usr/local/netscaler/samples
   Note: To uninstall the NSWL package, execute the following command:
   `rpm -e NSweblog`
   Note: To get more information about the NSweblog RPM file, execute the following command:
   `rpm -qpi *.rpm`
   Note: To view the installed Web server logging files, execute the following command:
   `rpm -qpl *.rpm`

Updated: 2014-06-25

To install the NSWL client, perform the following operations on the system where you downloaded the package.

1. Extract the nswl_bsd-<release number>-<build number>.tgz file from the package.
2. Copy the extracted file to a system, running FreeBSD OS, on which you want to install the NSWL client.
3. To install the NSWL package, execute the following command:
   `pkg_add nswl_bsd-9.3-51.5.tgz`
   This command extracts the files and installs them in the following directories.
   - /usr/local/netscaler/etc
   - /usr/local/netscaler/bin
   - /usr/local/netscaler/samples
   Note: To uninstall the NSWL package, execute the following command:
   `pkg_delete NSweblog`

4. To verify that the package is installed, execute the following command:
   pkg_info | grep NSweblog

Updated: 2014-06-25

To install the NSWL client, perform the following operations on the system where you downloaded the package.

1. Extract the nswl_macos-<release number>-<build number>.tgz file from the package.
2. Copy the extracted file to a system, running Mac OS, on which you want to install the NSWL client.
3. To install the NSWL package, execute the following command:
   pkg_add nswl_macos-9.3-51.5.tgz
   This command extracts the files and installs them in the following directories:
   - /usr/local/netscaler/etc
   - /usr/local/netscaler/bin
   - /usr/local/netscaler/samples
   Note: To uninstall the NSWL package, execute the following command:
   pkg_delete NSweblog
4. To verify that the package is installed, execute the following command:
   pkg_info | grep NSweblog

Updated: 2014-09-18

To install the NSWL client, perform the following operations on the system where you downloaded the package.

1. Extract the nswl_win-<release number>-<build number>.zip file from the package.
2. Copy the extracted file to a Windows system on which you want to install the NSWL client.
3. On the Windows system, unzip the file in a directory (referred as <NSWL-HOME>). The following directories are extracted: bin, etc, and samples.
4. At the command prompt, run the following command from the <NSWL-HOME>\bin directory:
   nswl -install -f <directorypath>\log.conf
   where,

   <directorypath> refers to the path of the configuration file (log.conf). By default, the file is in the <NSWL-HOME>\etc directory. However, you can copy the configuration file to any other directory.

   Note: To uninstall the NSWL client, at the command prompt, run the following command from the <NSWL-HOME>\bin directory:
   > nswl -remove

Updated: 2014-06-25

To install the NSWL client, perform the following operations on the system where you downloaded the package.

1. Extract the nswl_aix-<release number>-<build number>.rpm file from the package.
2. Copy the extracted file to a system, running AIX OS, on which you want to install the NSWL client.
3. To install the NSWL package, execute the following command:

```
rpm -i nswl_aix-9.3-51.5.rpm
```

This command extracts the files and installs them in the following directories.

- /usr/local/netscaler/etc
- /usr/local/netscaler/bin
- /usr/local/netscaler/samples

Note: To uninstall the NSWL package, execute the following command:

```
rpm -e NSweblog
```

Note: To get more information about the NSweblog RPM file, execute the following command:

```
rpm -qpi *.rpm
```

Note: To view the installed Web server logging files, execute the following command:

```
rpm -qpl *.rpm
```

Configuring the NSWL Client

Jun 01, 2015

After installing the NSWL client, you can configure the NSWL client using the nswl executable. These configurations are then stored in the NSWL client configuration file (log.conf).

Note: You can further customize logging on the NSWL client system by making additional modifications to the NSWL configuration file (log.conf). For details, see Customizing Logging on the NSWL Client System.
The following table describes the commands that you can use to configure the NSWL client.

NSWL command	Specifies
nswl -help	The available NSWL help options.
nswl -addns -f <path-to-configuration-file>	The system that gathers the log transaction data. You are prompted to enter the IP address of the NetScaler appliance. Enter a valid user name and password.
nswl -verify -f <path-to-configuration-file>	Check for syntax or semantic errors in the configuration file.
nswl -start -f <path-to-configuration-file>	Start the NSWL client based on the settings in the configuration file. Note: For Solaris and Linux: To start Web server logging as a background process, type the ampersand sign (&) at the end of the command.
nswl -stop (Solaris and Linux only)	Stop the NSWL client if it was started as a background process; otherwise, use CTRL+C to stop Web server logging.
nswl -install -f <path-to-configuration-file> (Windows only)	Install the NSWL client as a service in Windows.
nswl -startservice (Windows only)	Start the NSWL client by using the settings in the configuration file specified in the nswl install option. You can also start NSWL client from Start > Control Panel > Services.
nswl -stopservice (Windows only)	Stops the NSWL client.
nswl -remove	Remove the NSWL client service from the registry.

Run the following commands from the directory in which the NSWL executable is located:

- Windows: \ns\bin
- Solaris and Linux: \usr\local\netscaler\bin

The Web server logging configuration files are located in the following directory path:

- Windows: \ns\etc
- Solaris and Linux: \usr\local\netscaler\etc

The NSWL executable is started as .\nswl in Linux and Solaris.

This document includes the following details:

- Adding the IP Addresses of the NetScaler Appliance
- Verifying the NSWL Configuration File
- Running the NSWL Client

Updated: 2013-07-17

In the NSWL client configuration file (log.conf), add the NetScaler IP address (NSIP) from which the NSWL client will start collecting logs.

To add the NSIP address of the NetScaler appliance

1. At the client system command prompt, type:

   nswl -addns -f < directorypath > \log.conf

   < directorypath >: Specifies the path to the configuration file (log.conf).

2. At the next prompt, enter the following information:

   - **NSIP**: Specify the IP address of the NetScaler appliance.
   - Username and Password: Specify the nsroot user credentials of the NetScaler appliance.

Note: If you add multiple NetScaler IP addresses (NSIP), and later you do not want to log all of NetScaler system log details, you can delete the NSIPs manually by removing the NSIP statement at the end of the log.conf file. During a failover setup, you must add both primary and secondary NetScaler IP addresses to the log.conf by using the command. Before adding the IP address, make sure the user name and password exist on the NetScaler appliances.

Updated: 2013-06-20

To make sure that logging works correctly, check the NSWL configuration file (log.conf) on the client system for syntax errors.

To verify the configuration in the NSWL configuration file

At the client system command prompt, type:

nswl -verify -f <directorypath>\log.conf

< directorypath >: Specifies the path to the configuration file (log.conf).

Updated: 2013-06-20

To start Web server logging

At the client system command prompt, type:

nswl -start -f <directorypath>\log.conf

<directorypath>: Specifies the path to the configuration file (log.conf).

To stop Web server logging started as a background process on the Solaris or Linux operating systems

At the command prompt, type:

nswl -stop

To stop Web server logging started as a service on the Windows operating system

At the command prompt, type:

nswl -stopservice

Customizing Logging on the NSWL Client System

Jun 01, 2015

You can customize logging on the NSWL client system by making additional modifications to the NSWL client configuration file (log.conf). Use a text editor to modify the log.conf configuration file on the client system.

To customize logging, use the configuration file to define filters and log properties.

- **Log filters.** Filter log information based on the host IP address, domain name, and host name of the Web servers.
- **Log properties.** Each filter has an associated set of log properties. Log properties define how to store the filtered log information.

This document includes the following details:

Following is a sample configuration file:

```
##########
# This is the NSWL configuration file
# Only the default filter is active
# Remove leading # to activate other filters
##########
##########
# Default filter (default on)
# W3C Format logging, new file is created every hour or on reaching 10MB file size,
# and the file name is Exyymmdd.log
##########
Filter default
begin default
      logFormat            W3C
      logInterval          Hourly
      logFileSizeLimit      10
      logFilenameFormat     Ex%{%y%m%d}t.log
end default
##########
# netscaler caches example
# CACHE_F filter covers all the transaction with HOST name www.netscaler.com and the listed server ip's
##########
#Filter CACHE_F HOST www.netscaler.com IP 192.168.100.89 192.168.100.95 192.168.100.52 192.168.100.53 ON
##########
# netscaler origin server example
# Not interested in Origin server to Cache traffic transaction logging
##########
#Filter ORIGIN_SERVERS IP 192.168.100.64 192.168.100.65 192.168.100.66 192.168.100.67 192.168.100.225 192.168.100.226 192.168.
100.227 192.168.100.228 OFF
##########
# netscaler image server example
# all the image server logging.
##########
#Filter IMAGE_SERVER HOST www.netscaler.images.com IP 192.168.100.71 192.168.100.72 192.168.100.169 192.168.100.170 192.168.10
0.171 ON
##########
# NCSA Format logging, new file is created every day midnight or on reaching 20MB file size,
# and the file name is /datadisk5/netscaler/log/NS<hostname>/Nsmmddyy.log.
# Exclude objects that ends with .gif .jpg .jar.
```

```
###########
#begin ORIGIN_SERVERS
#       logFormat             NCSA
#       logInterval           Daily
#       logFileSizeLimit      40
#       logFilenameFormat     /datadisk5/ORGIN/log/%v/NS%{%m%d%y}t.log
#       logExclude            .gif .jpg .jar
#end ORIGIN_SERVERS

###########
# NCSA Format logging, new file is created every day midnight or on reaching 20MB file size,
# and the file name is /datadisk5/netscaler/log/NS<hostname>/Nsmmddyy.log with log record timestamp as GMT.
###########
#begin CACHE_F
#       logFormat             NCSA
#       logInterval           Daily
#       logFileSizeLimit      20
#       logFilenameFormat /datadisk5/netscaler/log/%v/NS%{%m%d%y}t.log
#       logtime               GMT
#end CACHE_F

###########
# W3C Format logging, new file on reaching 20MB and the log file path name is
# atadisk6/netscaler/log/server's ip/Exmmyydd.log with log record timestamp as LOCAL.
###########
#begin IMAGE_SERVER
#       logFormat             W3C
#       logInterval           Size
#       logFileSizeLimit      20
#       logFilenameFormat /datadisk6/netscaler/log/%AEx%{%m%d%y}t
#       logtime               LOCAL
#end IMAGE_SERVER

###########
# Virtual Host by Name firm, can filter out the logging based on the host name by,
###########

#Filter VHOST_F IP 10.101.2.151 NETMASK 255.255.255.0
#begin VHOST_F
#       logFormat             W3C
#       logInterval           Daily
#       logFileSizeLimit      10
logFilenameFormat /ns/prod/vhost/%v/Ex%{%m%d%y}t
#end VHOST_F

########## END FILTER CONFIGURATION ##########
```

Updated: 2014-01-06

You can use the default filter definition located in the configuration file (log.conf), or you can modify the filter or create a new filter. You can create more than one log filter.

Note: Consolidated logging, which logs transactions for which no filter is defined, uses the default filter if it is enabled. Consolidated logging of all servers can be done by defining only the default filter.
If the server hosts multiple Web sites and each Web site has its own domain name, and each domain is associated with a virtual server, you can configure Web server logging to create a separate log directory for each Web site. The following table displays the parameters for creating a filter.

Table 1. Parameters for Creating a Filter

Parameter	Specifies
filterName	Name of the filter. The filter name can include alphanumeric characters and cannot be longer than 59 characters. Filter names longer than 59 characters are truncated to 59 characters.

Parameter	Specifies
HOST name	Host name of the server for which the transactions are being logged.
IP ip	IP address of the server for which transactions are to be logged (for example, if the server has multiple domains that have one IP address).
IP ip 2...ip n:	Multiple IP addresses (for example, if the server domain has multiple IP addresses).
ip6 ip	IPv6 address of the server for which transactions are to be logged.
IP ip NETMASK mask	IP addresses and netmask combination to be used on a subnet.
ON | OFF	Enable or disable the filter to log transactions. If no argument is selected, the filter is enabled (ON).

To create a filter

To create a filter, enter the following command in the log.conf file:

- filter <filterName> <HOST name> | [IP<ip>] | [IP<ip 2...ip n>] | <IP ip NETMASK mask> [ON | OFF]
- filter <filterName> <HOST name> | [IP6 ip/<prefix length>] [ON | OFF]

To create a filter for a virtual server

To create a filter for a virtual server, enter the following command in the log.conf file:

filter <filterName> <VirtualServer IP address>

Example

In the following example, you specify an IP address of 192.168.100.0 and netmask of 255.255.255.0. The filter applies to IP addresses 192.168.100.1 through 192.168.100.254.

```
Filter F1 HOST www.netscaler.com ON
Filter F2 HOST www.netscaler.com IP 192.168.100.151 ON
Filter F3 HOST www.netscaler.com IP 192.168.100.151 192.165.100.152 ON
Filter F4 IP 192.168.100.151
Filter F5 IP 192.168.100.151 HOST www.netscaler.com OFF
Filter F6 HOST www.netscaler.com HOST www.xyz.com HOST www.abcxyz.com IP 192.168.100.200 ON
Filter F7 IP 192.250.100.0 NETMASK 255.255.255.0
Filter F8 HOST www.xyz.com IP 192.250.100.0 NETMASK 255.255.255.0 OFF
For creating filters for servers having IPv6 addresses.
Filter F9 2002::8/112 ON
Filter F10 HOST www.abcd.com IP6 2002::8 ON
```

Log properties are applied to all log entries associated with the filter. The log property definition begins with the keyword BEGIN and ends with END as illustrated in the following example:

```
BEGIN <filtername>
 logFormat ...
 logFilenameFormat ...
 logInterval ...
 logFileSize ....
 logExclude ....
 logTime ....
 END
```

Entries in the definition can include the following:

- **LogFormat** specifies the Web server logging feature that supports NCSA, W3C Extended, and custom log file formats.

  By default, the logformat property is w3c. To override, enter custom or NCSA in the configuration file, for example:

```
LogFormat NCSA
```

Note: For the NCSA and custom log formats, local time is used to time stamp transactions and for file rotation.

- **LogInterval** specifies the intervals at which new log files are created. Use one of the following values:
  - Hourly: A file is created every hour.
  - Daily: A file is created every day at midnight. Default value.
  - Weekly: A file is created every Sunday at midnight.
  - Monthly: A file is created on the first day of the month at midnight.
  - None: A file is created only once, when Web server logging starts.

  Example

  ```
  LogInterval Daily
  ```
- **LogFileSizeLimit** specifies the maximum size of the log file in MB. It can be used with any log interval (weekly, monthly, and so on.) A file is created when the maximum file size limit is reached or when the defined log interval time elapses.

  To override this behavior, specify the size as the loginterval property so that a file is created only when the log file size limit is reached.

  The default LogFileSizeLimit is 10 MB.

  Example

  ```
  LogFileSizeLimit 35
  ```
- **LogFilenameFormat** specifies the file name format of the log file. The name of the file can be of the following types:
  - Static: Specifies a constant string that contains the absolute path and file name.
  - Dynamic: Specifies an expression containing the following format:
    - Server IP address (%A)
    - Date (%{format}t)
    - URL suffix (%x)
    - Host name (%v)

  Example

  ```
  LogFileNameFormat Ex%{%m%d%y}t.log
  ```
  This command creates the first file name as Exmmddyy.log, then every hour creates a file with file name: Exmmddyy.log.0, Exmmddyy.log.1,..., Exmmddyy.log.n.

  Example

  ```
  LogInterval size
  LogFileSize 100
  LogFileNameFormat Ex%{%m%d%y}t
  ```
  Caution: The date format %t specified in the LogFilenameFormat command overrides the log interval property for that filter. To prevent a new file being created every day instead of when the specified log file size is reached, do not use %t in the LogFilenameFormat.
- **LogExclude** prevents logging of transactions with the specified file extensions.

  Example

  ```
  LogExclude .html
  ```
  This command creates a log file that excludes log transactions for *.html files.
- **LogTime** specifies log time as either GMT or LOCAL.

  The defaults are:
  - NCSA log file format: LOCAL
  - W3C log file format: GMT.

Updated: 2013-09-30

The NetScaler supports the following standard log file formats:

- NCSA Common Log Format
- W3C Extended Log Format

NCSA Common Log Format

If the log file format is NCSA, the log file displays log information in the following format:

```
Client_IP_address -User_Name [Date:Time -TimeZone] "Method Object HTTP_version"
HTTP_StatusCode BytesSent
```

To use the NCSA Common log format, enter NCSA in the LogFormat argument in the log.conf file.

The following table describes the NCSA Common log format.

Table 2. NCSA Common Log Format

Argument	Specifies
Client _IP_address	The IP address of the client computer.
User Name	The user name.
Date	The date of the transaction.
Time	The time when the transaction was completed.
Time Zone	The time zone (Greenwich Mean Time or local time).
Method	The request method (for example; GET, POST).
Object	The URL.
HTTP_version	The version of HTTP used by the client.
HTTP_StatusCode	The status code in the response.
Bytes Sent	The number of bytes sent from the server.

W3C Extended Log Format

An extended log file contains a sequence of lines containing ASCII characters terminated by either a Line Feed (LF) or the sequence Carriage Return Line Feed (CRLF.) Log file generators must follow the line termination convention for the platform on which they are run.

Log analyzers must accept either LF or CRLF form. Each line may contain either a directive or an entry. If you want to use the W3C Extended log format, enter W3C as the Log-Format argument in the log.conf file.

By default, the standard W3C log format is defined internally as the custom log format, shown as follows:

```
%{%Y-%m-%d%H:%M:%S}t %a %u %S %A %p %m %U %q %s %j %J %T %H %+{user-agent}i %+{cookie} i%+{referer}i
```

For a description of the meaning of this each custom format, see "Appendix A: Arguments for Defining a Custom Log Format." You can also change the order or remove some fields in this W3C log format. For example:

```
logFormat W3C %{%Y-%m-%d%H:%M:%S}t %m %U
```

W3C log entries are created with the following format:

```
#Version: 1.0
#Fields: date time cs-method cs-uri
#Date: 12-Jun-2001 12:34
2001-06-12 12:34:23 GET /sports/football.html
2001-06-12 12:34:30 GET /sports/football.html
```

Entries

Entries consist of a sequence of fields relating to a single HTTP transaction. Fields are separated by white space; Citrix recommends the use of tab characters. If a field in a particular entry is not used, a dash (-) marks the omitted field.

Directives

Directives record information about the logging process. Lines beginning with the pound sign (#) contain directives.

The following table describes the directives.

Table 3. Directive Descriptions

Directive	Description
Version: <integer>.<integer>	Displays the version of the extended log file format used. This document defines version 1.0.
Fields: [<specifier>...]	Identifies the fields recorded in the log.
Software: <string>	Identifies the software that generated the log.
Start-Date: <date> <time>	Displays the date and time at which the log was started.
End-Date: <date> <time>	Displays the date and time at which logging finished.
Date: <date> <time>	Displays the date and time when the entry was added.
Remark: <text>	Displays comments. Analysis tools ignore data recorded in this field.

Note: The Version and Fields directives are required. They precede all other entries in the log file.

Example

The following sample log file shows the log entries in W3C Extended log format:

```
#Version: 1.0
#Fields: time cs-method cs-uri
#Date: 12-Jan-1996 00:00:00
00:34:23 GET /sports/football.html
12:21:16 GET /sports/football.html
12:45:52 GET /sports/football.html
12:57:34 GET /sports/football.html
```

Fields

The Fields directive lists a sequence of field identifiers that specify the information recorded in each entry. Field identifiers may have one of the following forms:

- **identifier:** Relates to the transaction as a whole.
- **prefix-identifier:** Relates to information transfer between parties defined by the value prefix.
- **prefix (header):** Specifies the value of the HTTP header field header for transfer between parties defined by the value prefix. Fields specified in this manner always have the type <string>.

The following table describes defined prefixes.

Table 4. Prefix Descriptions

Prefix	Specifies
c	Client
s	Server
r	Remote
cs	Client to server

Prefix	Specifies
sc	Server to client
sr	Server to remote server (prefix used by proxies)
rs	Remote server to server (prefix used by proxies)
x	Application-specific identifier

Examples

The following examples are defined identifiers that use prefixes:

cs-method: The method in the request sent by the client to the server.

sc(Referer): The Referer field in the reply.

c-ip: The IP address of the client.

Identifiers

The following table describes the W3C Extended log format identifiers that do not require a prefix.

Table 5. W3C Extended Log Format Identifiers (No Prefix Required)

Identifier	Description
date	The date on which the transaction was done.
time	The time when the transaction is done.
time-taken	The time taken (in seconds) for the transaction to complete.
bytes	The number of bytes transferred.
cached	Records whether a cache hit has occurred. A zero indicates a cache miss.

The following table describes the W3C Extended log format identifiers that require a prefix.

Table 6. W3C Extended Log Format Identifiers (Requires a Prefix)

Identifier	Description
IP	The IP address and the port number.
dns	The DNS name.
status	The status code.
comment	The comment returned with status code.
method	The method.
url	The URL.
url-stem	The stem portion of the URL.

Identifier	Description
cs-uri-query	The query portion of the URL.

The W3C Extended Log file format allows you to choose log fields. These fields are shown in the following table.

Table 7. W3C Extended Log File Format (Allows Log Fields)

Field	Description
Date	The date on which the transaction is done.
Time	The time when the transaction is done.
Client IP	The IP address of the client.
User Name	The user name.
Service Name	The service name, which is always HTTP.
Server IP	The server IP address.
Server Port	The server port number
Method	The request method (for example; GET, POST).
Url Stem	The URL stem.
Url Query	The query portion of the URL.
Http Status	The status code in the response.
Bytes Sent	The number of bytes sent to the server (request size, including HTTP headers).
Bytes Received	The number of bytes received from the server (response size, including HTTP headers).
Time Taken	The time taken for transaction to complete, in seconds.
Protocol Version	The version number of HTTP being used by the client.
User Agent	The User-Agent field in the HTTP protocol.
Cookie	The Cookie field of the HTTP protocol.
Referer	The Referer field of the HTTP protocol.

Updated: 2013-09-30

You can customize the display format of the log file data manually or by using the NSWL library. By using the custom log format, you can derive most of the log formats that Apache currently supports.

Creating a Custom Log Format by Using the NSWL Library

Use one of the following NSWL libraries depending on whether the NSWL executable has been installed on a Windows or Solaris host computer:

- **Windows:** The nswl.lib library located in \ns\bin directory on the system manager host computer.
- **Solaris:** The libnswl.a library located in /usr/local/netscaler/bin.

1. Add the following two C functions defined by the system in a C source file:
   ns_userDefFieldName() : This function returns the string that must be added as a custom field name in the log record.

   ns_userDefFieldVal() : This function implements the custom field value, then returns it as a string that must be added at the end of the log record.
2. Compile the file into an object file.
3. Link the object file with the NSWL library (and optionally, with third party libraries) to form a new NSWL executable.
4. Add a %d string at the end of the logFormat string in the configuration file (log.conf).

Example

```
##########
# A new file is created every midnight or on reaching 20MB file size,
# and the file name is /datadisk5/netscaler/log/NS<hostname>/Nsmmddyy.log and create digital
#signature field for each record.
BEGIN CACHE_F
 logFormat  custom "%a - "%{user-agent}i" [%d/%B/%Y %T -%g] "%x" %s %b%{referrer}i "%{user-agent}i" "%{cookie}i" %d "
 logInterval  Daily
 logFileSizeLimit  20
 logFilenameFormat  /datadisk5/netscaler/log/%v/NS%{%m%d%y}t.log
END CACHE_F
```

Creating a Custom Log Format Manually

To customize the format in which log file data should appear, specify a character string as the argument of the LogFormat log property definition. For more information, see "Appendix A: Arguments for Defining a Custom Log Format." The following is an example where character strings are used to create a log format:

```
 LogFormat Custom ""%a - "%{user-agent}i" %[%d/%m/%Y]t %U %s %b %T"
```

- The string can contain the “c” type control characters \n and \t to represent new lines and tabs.
- Use the <Esc> key with literal quotes and backslashes.

The characteristics of the request are logged by placing % directives in the format string, which are replaced in the log file by the values.

If the %v (Host name) or %x (URL suffix) format specifier is present in a log file name format string, the following characters in the file name are replaced by an underscore symbol in the log configuration file name:

" * . / : < > ? \ |

Characters whose ASCII values lie in the range of 0-31 are replaced by the following:

%<ASCII value of character in hexadecimal>.

For example, the character with ASCII value 22 is replaced by %16.

Caution: If the %v format specifier is present in a log file name format string, a separate file is opened for each virtual host. To ensure continuous logging, the maximum number of files that a process can have open should be sufficiently large. See your operating system documentation for a procedure to change the number of files that can be opened.

Creating Apache Log Formats

You can derive from the custom logs most of the log formats that Apache currently supports. The custom log formats that match Apache log formats are:

NCSA/combined: LogFormat custom %h %l %u [%t] "%r" %s %B "%{referer}i" "%{user-agent}i"

NCSA/Common: LogFormat custom %h %l %u [%t] "%r" %s %B

Referer Log: LogFormat custom "%{referer}i" -> %U

Useragent: LogFormat custom %{user-agent}i

Similarly, you can derive the other server log formats from the custom formats.

Updated: 2015-04-02

The following table describes the data that you can use as the Log Format argument string:

Table 8. Custom Log Format

Argument	Specifies
%a	Remote IPv4 address.
%A	Local IPv4 address.
%a6	Remote IPv6 address.
%A6	Local IPv6 address.
%B	Bytes sent, excluding the HTTP headers (response size).
%b	Bytes received, excluding the HTTP headers (request size).
%d	User-defined field.
%e1	Value of the first custom HTTP request header.
%e2	Value of the second custom HTTP request header.
%E1	Value of the first custom HTTP response header.
%E2	Value of the second custom HTTP response header.
Note: For instructions on how to export custom HTTP headers, see "Configuring the NetScaler for Web Server Logging."	
%g	Greenwich Mean Time offset (for example, -0800 for Pacific Standard Time).
%h	Remote host.
%H	Request protocol.
%{Foobar}i	Contents of the Foobar: header line(s) in the request sent to the server. The system supports the User-Agent, Referer and cookie headers. The + after the % in this format informs the logging client to use the + as a word separator.
%j	Bytes received, including headers (request size)
%J	Bytes sent, including headers (response size)
%l	Remote log name (from identd, if supplied).
%m	Request method.
%M	Time taken to serve the request (in microseconds)
%	Contents of Foobar: header line(s) in the reply. USER-AGENT, Referer, and cookie headers (including set cookie headers) are supported.

Argument	Specifies
{Foobar}o	
%p	Canonical port of the server serving the request.
%q	Query string (prefixed with a question mark (?) if a query string exists).
%r	First line of the request.
%s	Requests that were redirected internally, this is the status of the original request.
%t	Time, in common log format (standard English time format).
%{format}t	Time, in the form given by format, must be in the strftime(3) format. For format descriptions, see "Appendix B: Time Format Definition."
%T	Time taken to serve the request, in seconds.
%u	Remote user (from auth; may be bogus if return status (%s) is 401).
%U	URL path requested.
%v	Canonical name of the server serving the request.
%V	Virtual server IPv4 address in the system, if load balancing, content switching, and/or cache redirection is used.
%V6	Virtual server IPv6 address in the system, if load balancing, content switching, and/or cache redirection is used.

For example, if you define the log format as %+{user-agent}i, and if the user agent value is Citrix NetScaler system Web Client, then the information is logged as NetScaler system+Web+Client. An alternative is to use double quotation marks. For example, "%{user-agent}i" logs it as "Citrix NetScaler system Web Client." Do not use the <Esc> key on strings from %.. .r, %. . .i and, %. . .o. This complies with the requirements of the Common Log Format. Note that clients can insert control characters into the log. Therefore, you should take care when working with raw log files.

Updated: 2015-04-28

The following table lists the characters that you can enter as the format part of the %{format}t string described in the Custom Log Format table of "Arguments for Defining a Custom Log Format." Values within brackets ([ ]) show the range of values that appear. For example, [1,31] in the %d description in the following table shows %d ranges from 1 to 31.

Table 9. Time Format Definition

Argument	Specifies
%%	The same as %.
%a	The abbreviated name of the week day for the locale.
%A	The full name of the week day for the locale.
%b	The abbreviated name of the month for the locale.
%B	The full name of the month for the locale.
%C	The century number (the year divided by 100 and truncated to an integer as a decimal number [1,99]); single digits are preceded by a 0.

Argument	Specifies
%d	The day of month [1,31]; single digits are preceded by 0.
%e	The day of month [1,31]; single digits are preceded by a blank.
%h	The abbreviated name of the month for the locale.
%H	The hour (24-hour clock) [0,23]; single digits are preceded by a 0.
%I	The hour (12-hour clock) [1,12]; single digits are preceded by a 0.
%j	The number of the day in the year [1,366]; single digits are preceded by 0.
%k	The hour (24-hour clock) [0,23]; single digits are preceded by a blank.
%l	The hour (12-hour clock) [1,12]; single digits are preceded by a blank.
%m	The number of the month in the year [1,12]; single digits are preceded by a 0.
%M	The minute [00,59]; leading 0 is permitted but not required.
%n	Inserts a new line.
%p	The equivalent of either a.m. or p.m. for the locale.
%r	The appropriate time representation in 12-hour clock format with %p.
%S	The seconds [00,61]; the range of values is [00,61] rather than [00,59] to allow for the occasional leap second and for the double leap second.
%t	Inserts a tab.
%u	The day of the week as a decimal number [1,7]. 1 represents Sunday, 2 represents Tuesday and so on.
%U	The number of the week in the year as a decimal number [00,53], with Sunday as the first day of week 1.
%w	The day of the week as a decimal number [0,6]. 0 represents Sunday.
%W	Specifies the number of the week in the year as a decimal number [00,53]. Monday is the first day of week 1.
%y	The number of the year within the century [00,99]. For example, 5 would be the fifth year of that century.
%Y	The year, including the century (for example, 1993).

Note: If you specify a conversion that does not correspond to any of the ones described in the preceding table, or to any of the modified conversion specifications listed in the next paragraph, the behavior is undefined and returns 0.
The difference between %U and %W (and also between modified conversions %OU and %OW) is the day considered to be the first day of the week. Week number 1 is the first week in January (starting with a Sunday for %U, or a Monday for %W). Week number 0 contains the days before the first Sunday or Monday in January for %U and %W.

Configuring Call Home

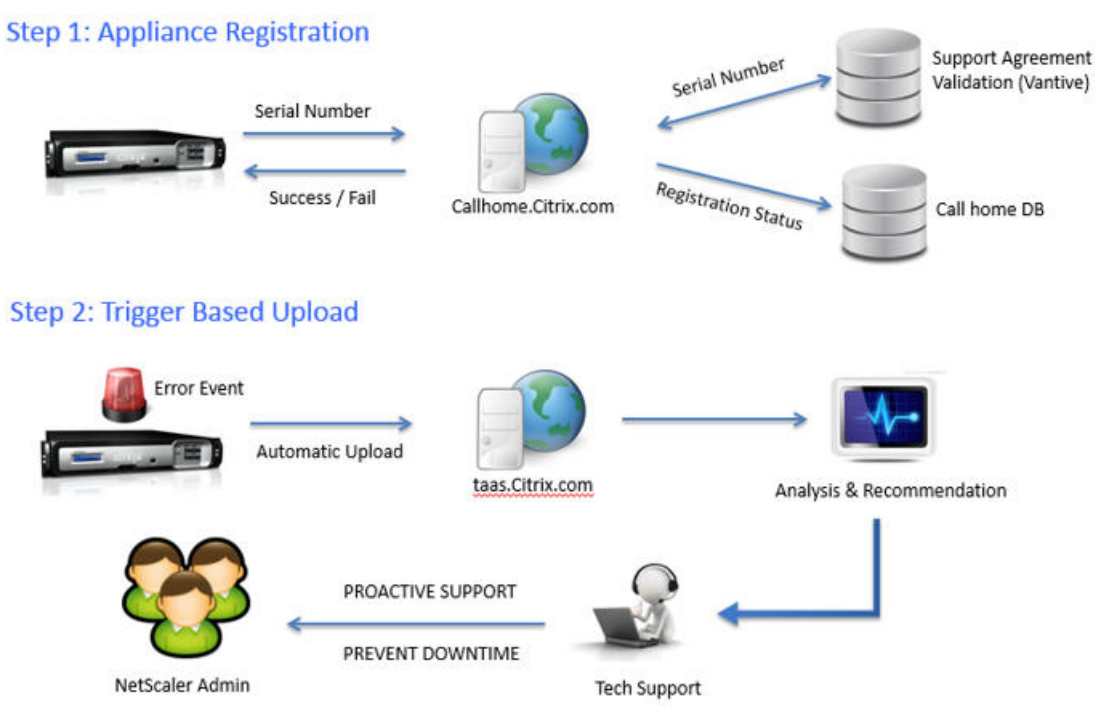
Jun 14, 2016

The Call Home feature monitors your NetScaler appliance for critical error conditions. Call Home registers your appliance with the Citrix Technical Support server. If your appliance is successfully registered with the Support server, Call Home automatically uploads system data to that server in the event that one of the conditions occurs. The NetScaler Appliance keeps a full log of all upload events. If you are unable to correct the problem after reviewing the appliance's log, you can contact the Citrix Technical Support team and open a service request. The team can analyze the uploaded system data and recommend possible solutions.

The Call Home feature is supported on all three platforms of NetScaler ADC.

- In NetScaler MPX, Call Home feature is supported on all MPX models.
- In NetScaler SDX, Call Home feature is supported on all VPX instances running on a SDX box 002E

Following is a typical set up for Call Home.



The process flow for using Call Home can be categorized as follows:

- Registration of the NetScaler appliance to the Citrix Technical Support server.
- Uploading of the appliance's data to the Citrix Technical Support server. The support server has the following URL: https://callhome.citrix.com/.
- Opening a Technical Support case (Optional).

By default, the Citrix Call Home service is disabled on the appliance. You must first enable the service to register the appliance for critical error conditions.

Registration of the NetScaler appliance. The appliance has to be registered to the Citrix Technical Support server before Call Home can upload the system data to the server when predefined error conditions occur on the appliance. Enabling the Call Home feature on the NetScaler appliance initiates the registration process. The process flow is as follows:

1. The Call Home process sends the following details to the Citrix Technical Support server:

- Hardware serial number is shared for NetScaler MPX and SDX models
- License serial number is shared for NetScaler VPX models.

2. The server checks its database for an active technical support service contract for the appliance.
3. If there is an active technical support service contract, the support server registers the NetScaler appliance for Call Home and sends a successful-registration response to the appliance stating that the feature is successfully enabled. If there is no active technical support service contract, the server sends a registration-failure response to the NetScaler appliance.

The following table lists the error conditions that Call Home currently monitors on a NetScaler Appliance:

Table 1. List of error conditions monitored by Call Home

Error Condition	Indicates	Call Home Monitoring Interval	Corresponding SNMP Alarm Name
Compact flash drive errors	The compact flash drive on the appliance that encountered read or write errors.	24 hours	COMPACT-FLASH-ERRORS
Hard disk drive errors	The hard drives on the appliance that encountered read or write errors.	24 hours	HARD-DISK-DRIVE-ERRORS
Power supply unit failure	One of the power supply units on the NetScaler appliance has failed.	7 seconds	POWER-SUPPLY-FAILURE
SSL card failure	One of the SSL cards on the NetScaler appliance has failed.	7 seconds	SSL-CARD-FAILED
Warm restart	The appliance has warm restarted due to a failure of a system process.	After every restart of the NetScaler appliance.	WARM-RESTART-EVENT

Note: The Call Home feature do not monitor the power supply unit (PSU) status for VPX models and VPX instances.

Uploading of appliance's data to the Support server. An error condition triggers the following sequence of events:

1. The Call Home process checks the registration status. If the status indicates successful registration, the process advances to the next step.
2. The Call Home process runs a showtech support script that collects all of the system related data in a tar file. The data in the tar file includes configurations, logs, and statistics. Call Home locally saves the tar file at /var/tmp/support/callhome.
3. Call Home uploads a copy of the tar file to the Citrix Technical Support server. The Appliance logs the uploading of the tar file in a log file named *callhome.log* located at /var/log. You can also configure the CALLHOME-UPLOAD-EVENT SNMP alarm to generate an SNMP alert whenever Call Home uploads happen.
4. If the SNMP alarm related to the error condition is enabled, the SNMP agent on the appliance generates an SNMP trap message and sends it to all of the configured SNMP trap destinations. For more information about configuring SNMP alarms and trap destinations, see "Configuring the NetScaler to Generate SNMPv1 and SNMPv2 Traps."

Note: Call Home creates the Call Home tar file and uploads it to the CITRIX tech support server for only the first occurrence of a particular error condition since the appliance was last restarted. If you want the NetScaler appliance to send you alerts each time a particular error condition occurs, configure the corresponding SNMP alarm for the error condition.

The Call Home tar file has the following name format:

collector_callhome_<NSIP of the appliance>_<P for Primary or standalone, or S for Secondary>_<date>_<hours, in 24 hr format, according to the local time zone>_<minutes>.tar.gz. For example, collector_callhome_10.105.13.100_P_2Feb2012_20_30.tar.gz.

Opening a Technical Support Service Request . After you review the logs and SNMP trap messages for Call Home upload events, you have the option of contacting the Citrix Technical Support team and opening a service request. For more information about contacting the team and opening a service request, see http://support.citrix.com/article/CTX132307.

The Support team can then analyze the system data in the uploaded Call Home tar files and sends recommendations for possible solutions to the administrator's email address.

Before you begin configuring Call Home, do the following:

- Make sure that the NetScaler appliance is connected to the Interne or to a proxy server that has internet connectivity.
- Make sure that you have an active Citrix Technical Support service contract for the appliance.

Configuring Call Home on the NetScaler appliance consists of the following tasks:

1. **Enable the Call Home feature**. When you enable the Call Home feature, the Call Home process registers the appliance with the Citrix Technical Support server. The registration takes some time to complete. During that time, the appliance displays the status as IN PROGRESS. When the registration is complete, the appliance displays the status as SUCCESSFUL.
   Note: While upgrading the NetScaler appliance from an older release to release 10.1 or later, the NetScaler appliance prompts you to enable the Call Home feature, if:
   - The Call Home feature is not supported in the older release.
   - The Call Home feature is disabled in the older release.
2. **(Optional) Specify the administrator's email address**. The Call Home process sends the email address to the Support server, where it is stored for future correspondence regarding Call Home uploads.
3. **(Optional) Specify Proxy server settings**. Netscaler appliance needs internet connectivity to upload the collector archive to the Citrix Technical Support server. If the appliance does not have internet connectivity, then a proxy server (having internet connectivity) can be configured to upload the data.
4. **(Optional) Enable the CALLHOME-UPLOAD-EVENT SNMP alarm**. The SNMP agent on the NetScaler appliance generates a trap message and sends to all the configured SNMP trap destinations. The message includes the status of uploading of the Call Home tar file by the Call Home process. For more information about configuring SNMP alarms and trap destinations, see "Configuring the NetScaler to Generate SNMPv1 and SNMPv2 Traps."
5. **(Optional) Enable all of the corresponding SNMP alarms**. Call Home creates and uploads a Call Home tar file for the first occurrence of a monitored error condition since the appliance was last restarted. If you want to be alerted of these error conditions, you can configure the corresponding SNMP alarm. Table 1 lists all the corresponding SNMP alarms. For more information about configuring SNMP alarms and trap destinations, see "Configuring the NetScaler to Generate SNMPv1 and SNMPv2 Traps."

At the command prompt, type any of the following:

- enable ns feature ch
- enable ns feature callhome

At the command prompt, type:
show callhome
Example

```
> enable ns feature ch
Done

> show callhome
 Callhome feature: ENABLED
 Registration with Citrix upload server IN PROGRESS

 E-mail address configured:
 Proxy mode:NO   Ipaddress:     Port:0

 Trigger event              State     First occurrence   Latest occurrence
 -------------              -----     ----------------   -----------------
 1) Compact flash errors       Enabled   ..              ..
 2) Hard disk drive errors     Enabled   ..              ..
 3) Power supply unit failure   Enabled   ..              ..
 4) SSL card failure           Enabled   ..              ..
 5) Warm restart               Enabled   N/A             ..
 Done

 > show callhome
 Callhome feature: ENABLED
 Registration with Citrix upload server SUCCESSFUL

 E-mail address configured:
 Proxy mode:NO   Ipaddress:     Port:0

 Trigger event              State     First occurrence   Latest occurrence
 -------------              -----     ----------------   -----------------
 1) Compact flash errors       Enabled    ..              ..
 2) Hard disk drive errors     Enabled    ..              ..
 3) Power supply unit failure   Enabled    ..              ..
 4) SSL card failure           Enabled    ..              ..
 5) Warm restart               Enabled    N/A             ..
 Done
```

To specify the administrator's email address and proxy server settings by using the command line interface

At the command prompt, type:

- set callhome -emailAddress <string>
- set callhome -proxyMode (YES | NO) [-IPAddress <ip_addr|ipv6_addr|*>] [-port <port|*>]
- show callhome

Example

```
> set callhome -emailAddress exampleadmin@example.com
Done

> set callhome –proxyMode Yes –IPAddress 10.102.167.33 –port 80
Done

> show callhome
 E-mail address configured: exampleadmin@example.com
 Proxy mode:YES   Ipaddress: 10.102.167.33      Port:80

 Trigger event              State     First occurrence   Latest occurrence
 -------------              -----     ----------------   -----------------
 1) Compact flash errors      Enabled   ..                 ..
 2) Hard disk drive errors    Enabled   ..                 ..
 3) Power supply unit failure  Enabled   ..                 ..
 4) SSL card failure          Enabled   ..                 ..
 5) Warm restart              Enabled   N/A                ..
 Done
```

At the command prompt, type:

- set callhome -ipAddress <ipaddress> -port <port> -proxyMode [yes | no]
- show callhome
  Note: Proxy mode is enabled only when the -proxymode parameter is set to YES. If it is set to NO, the proxy functionality does not work, even if the IP address and port are configured. The port number should be for an HTTP service on the proxy server, not for an HTTPS service.

Example

```
> set callhome ipAddress 10.0.0.1 -port 80 -proxyMode yes
Done
```

Navigate to System > Settings, click Configure Advanced Features and select the Call Home option.

Navigation to System > Diagnostics and, in the Technical Support Tools group, select Call Home option to view the status of registration.

Navigation to System > Diagnostics and, in the Technical Support Tools group, select Call Home option to specify the administrator's email address.

Navigation to System > Diagnostics and, in the Technical Support Tools group, select Call Home option to specify the proxy server's IP address and the port number.

Reporting Tool

Jun 01, 2015

Use the Citrix® NetScaler® Reporting tool to view NetScaler performance statistics data as reports. Statistics data are collected by the nscollect utility and are stored in a database. When you want to view certain performance data over a period of time, the Reporting tool pulls out specified data from the database and displays them in charts.

Reports are a collection of charts. The Reporting tool provides built-in reports as well as the option to create custom reports. In a report, you can modify the charts and add new charts. You can also modify the operation of the data collection utility, nscollect, and stop or start its operation.

This document includes the following details:

- Using the Reporting Tool
- Working with Reports
- Working with Charts
- Examples
- Stopping and Starting the Data Collection Utility

The Reporting tool is a Web-based interface accessed from the Citrix® NetScaler® appliance. Use the Reporting tool to display the performance statistics data as reports containing graphs. In addition to using the built-in reports, you can create custom reports, which you can modify at any time. Reports can have between one and four charts. You can create up to 256 custom reports.

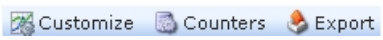
To invoke the Reporting tool

1. Use the Web browser of your choice to connect to the IP address of the NetScaler (for example, http://10.102.29.170/). The Web Logon screen appears.
2. In the User Name text box, type the user name assigned to the NetScaler.
3. In the Password text box, type the password.
4. In the Start in drop-down box, select Reporting.
5. Click Login.

The following screen shots show the report toolbar and the chart toolbar, which are frequently referenced in this documentation.

Figure 1. *Report Toolbar*

Figure 2. *Chart Toolbar*



Updated: 2013-09-27

You can plot and monitor statistics for the various functional groups configured on the NetScaler over a specified time interval. Reports enable you to troubleshoot or analyze the behavior of your appliance. There are two types of reports: built-in reports and custom reports. Report content for built-in or custom reports can be viewed in a graphical format or a

tabular format. The graphical view consists of line, area, and bar charts that can display up to 32 sets of data (counters). The tabular view displays the data in columns and rows. This view is useful for debugging error counters.

The default report that is displayed in the Reporting tool is CPU vs. Memory Usage and HTTP Requests Rate. You can change the default report view by displaying the report you want as your default view, and then clicking Default Report.

Reports can be generated for the last hour, last day, last week, last month, last year, or you can customize the duration.

You can do the following with reports:

- Toggle between a tabular view of data and a graphical view of data.
- Change the graphical display type, such as bar chart or line chart.
- Customize charts in a report.
- Export the chart as an Excel comma-separated value (CSV) file.
- View the charts in detail by zooming in, zooming out, or using a drag-and-drop operation (scrolling).
- Set a report as the default report for viewing whenever you log on.
- Add or remove counters.
- Print reports.
- Refresh reports to view the latest performance data.

Using Built-in Reports

The Reporting tool provides built-in reports for frequently viewed data. Built-in reports are available for the following functional groups: System, Network, SSL, Compression, Integrated Cache, NetScaler Gateway, and Citrix NetScaler Application Firewall. By default, the built-in reports are displayed for the last day. However, you can view the reports for the last hour, last week, last month, or last year.

Note: You cannot save changes to built-in reports, but you can save a modified built-in report as a custom report.

1. In the left pane of the Reporting tool, under Built-in Reports, expand a group (for example, SSL).
2. Click a report (for example, SSL > All Backend Ciphers).

Creating and Deleting Reports

You can create your own custom reports and save them with user-defined names for reuse. You can plot different counters for different groups based on your requirements. You can create up to 256 custom reports.

You can either create a new report or save a built-in report as a custom report. By default, a newly created custom report contains one chart named System Overview, which displays the CPU Usage counter plotted for the last day. You can customize the interval and set the data source and time zone from the report toolbar. Within a report, you can use the chart toolbars to add, modify, or delete charts, as described in "Working with Charts."

By default, newly created custom reports contain one chart named System Overview that displays a CPU Usage counter plotted for the last day.

1. In the Reporting tool, on the report toolbar, click Create, or if you want to create a new custom report based on an existing report, open the existing report, and then click Save As.
2. In Report Name box, type a name for the custom report.
3. Do one of the following:

- To add the report to an existing folder, in Create in or Save in, click the down arrow to choose an existing folder, and then click OK.
- To create a new folder to store the report, click the Click to add folder icon, in Folder Name, type the name of the folder, and in Create in, specify where you want the new folder to reside in the hierarchy, and then click OK.

Note: You can create up to 128 folders.

1. In the left pane of the Reporting tool, next to Custom Reports, click the Click to manage custom reports icon.
2. Select the check box that corresponds with the report you want to delete, and then click Delete.

Note: When you delete a folder, all the contents of that folder are deleted.

Modifying the Time Interval

By default, built-in reports display data for the last day. However, if you want to change the time interval for a built-in report, you can save the report as a custom report. The new interval applies to all charts in the report. The following table describes the time-interval options.

Table 1. Time Intervals

Time interval	Displays
Last Hour	Statistics data collected for the last hour.
Last Day	Statistics data collected for the last day (24 hours).
Last Week	Statistics data collected for the last week (7 days).
Last Month	Statistics data collected for the last month (31 days).
Last Year	Statistics data collected for the last year (365 days).
Custom	Statistics data collected for a time period that you are prompted to specify.

1. In the left pane of the Reporting tool, click a report.
2. On the report toolbar, click Duration, and then click a time interval.

Setting the Data Source and Time Zone

You can retrieve data from different data sources to display them in the reports. You can also define the time zone for the reports and apply the currently displayed report's time selection to all the reports, including the built-in reports.

1. In the Reporting tool, on the report toolbar, click Settings.
2. In the Settings dialog box, in Data Source, select the data source from which you want to retrieve the counter information.
3. Do one or both of the following:
   - If you want the tool to remember the time period for which a chart is plotted, select the Remember time selection for charts check box.
   - If you want the reports to use the time settings of your NetScaler appliance, select the Use Appliance's time zone check box.

Exporting and Importing Custom Reports

You can share reports with other NetScaler administrators by exporting reports. You can also import reports.

1. In the left pane of the Reporting tool, next to Custom Reports, click the Click to manage custom reports icon.
2. Select the check box that corresponds with the report you want to export or import, and then click Export or Import.
   Note: When you export the file, it is exported in a .gz file format.

Updated: 2013-09-06

Use charts to plot and monitor counters or groups of counters. You can include up to four charts in one report. In each chart, you can plot up to 32 counters. The charts can use different graphical formats (for example, area and bar). You can move the charts up or down within the report, customize the colors and visual display for each counter in a chart, and delete a chart when you do not want to monitor it.

In all report charts, the horizontal axis represents time and the vertical axis represents the value of the counter.

Adding a Chart

When you add a chart to a report, the System Overview chart appears with the CPU Usage counter plotted for the last one day. To plot a different group of statistics or select a different counter, see "Modifying a Chart."

Note: If you add charts to a built-in report, and you want to retain the report, you must save the report as a custom report. Use the following procedure to add a chart to a report.

1. In the left pane of the Reporting tool, click a report.
2. Under the chart where you want to add the new chart, click the Add icon.

Modifying a Chart

You can modify a chart by changing the functional group for which the statistics are displayed and by selecting different counters.

1. In the left pane of the Reporting tool, click a report.
2. Under the chart that you want to modify, click Counters.
3. In the dialog box that appears, in the Title box, type a name for the chart.
4. Next to Plot chart for, do one of the following:
   - To plot counters for global counters, such as Integrated Cache and Compression, click System global statistics.
   - To plot entity counters for entity types, such as Load Balancing and GSLB, click System entities statistics.
5. In Select group, click the desired entity.
6. Under Counters, in Available, click the counter name(s) that you want to plot, and then click the > button.
7. If you selected System entities statistics in step 4, on the Entities tab, under Available, click the entity instance name(s) you want to plot, and then click the > button.
8. Click OK.

Viewing a Chart

You can specify the graphical formats of the plotted counters in a chart. Charts can be viewed as line charts, spline charts, step-line charts, scatter charts, area charts, bar charts, stacked area charts, and stacked bar charts. You can also zoom in, zoom out, or scroll inside the plot area of a chart. You can zoom in or out for all data sources for 1 hour, 1 day, 1 week, 1 month, 1 year, and 3 years.

Other options for customizing the view of a chart include customizing the axes of the charts, changing the background and edge color of the plot area, customizing the color and size of the grids, and customizing the display of each data set (counter) in a chart.

Data set numbers, such as Data Set 1, correspond to the order in which the counters in your graph are displayed at the bottom of the chart. For example, if CPU usage and Memory usage are displayed in first and second order at the bottom of the chart, CPU usage is equal to Data Set 1 and Memory usage is equal to Data Set 2.

Whenever you modify a built-in report, you need to save the report as a custom report to retain your changes.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart you want to view, on the chart toolbar, click Customize.
3. On the Chart tab, under Category, click Plot type, and then click the graph type you want to display for the chart. If you want to display the graph is 3D, select the Use 3D check box.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, on the report toolbar, click Zoom In, and do one or both of the following:
   - To refocus the chart to display data for a specific time window, drag and drop the cursor from the start time to the end time. For example, you can view data for a one-hour period on a certain day.
   - To refocus the chart to display data for a data point, simply click once on chart where you want to zoom in and get more detailed information.
3. Once you have the desired range of time for which you want to view detailed data, on the report toolbar, click Tabular

View. Tabular view displays the data in numeric form in rows and columns.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, on the report toolbar, click Tabular View. To return to the graphical view, click Graphical View.
   Note: You can also view the numeric data in the graphical view by hovering your cursor over the notches in the gridlines.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, on the report toolbar, click Scroll, and then click inside the chart and drag the cursor in the direction for which you want to see data for a new time period. For example, if you want to view data in the past, click and drag to the left.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart for which you want to customize the axes, click Customize.
3. On the Chart tab, under Category, click one or more of the following:
   - To change the background color, click Background Color, and then select the options for color, transparency, and effects.
   - To change the text color, click Text Color, and then select the options for color, transparency, and effects.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart for which you want to customize the axes, click Customize.
3. On the Chart tab, under Category, click one or more of the following:
   - To change the scale of the left y-axis, click Left Y-Axis, and then select the scale you want.
   - To change the scale of the right y-axis, click Right Y-Axis, in Data set to plot, select the date set, and then select the scale you want.
     Note: The data set numbers, such as Data Set 1, correspond to the order in which the counters in your graph are displayed at the bottom of the chart. For example, if CPU usage and Memory usage are displayed in first and second order at the bottom of the chart, CPU usage is equal to Data Set 1 and Memory usage is equal to Data Set 2.
   - To plot each data set in its own hidden y-axis, click Multiple Axes, and then click Enable.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart for which you want to customize the plot area, click Customize.
3. On the Plot Area tab, under Category, click one or more of the following:
   - To change the background color and edge color of the chart, click Background Color and Edge Color, and then select the options for color, transparency, and effects.
   - To change the horizontal or vertical grids of the chart, click Horizontal Grids or Vertical Grids, and then select the options for displaying the grids, grid width, grid color, transparency, and effects.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart for which you want to customize the display of the data set (counters), click Customize.

3. On the Data Set tab, in Select Data Set, select the data set (counter) for which you want to customize the graphical display.
   Note: The data set numbers, such as Data Set 1, correspond to the order in which the counters in your graph are displayed at the bottom of the chart. For example, if CPU usage and Memory usage are displayed in first and second order at the bottom of the chart, CPU usage is equal to Data Set 1 and Memory usage is equal to Data Set 2.
4. Under Category, do one of more of the following:
   - To change the background color, click Color, and then select the options for color, transparency, and effects.
   - To change the graph type, click Plot type, and then select the graph type you want to display for the data set. If you want to display the graph as 3D, select the Use 3D check box.

For further data analysis, you can export charts to Excel in a comma-separated value (CSV) format.

To export chart data to Excel

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart with the data you want to export to Excel, click Export.

Deleting a Chart

If you do not want to use a chart, you can remove it from the report. You can permanently remove charts from custom reports only. If you delete a chart from a built-in report and want to retain the changes, you need to save the report as a custom report.

1. In the left pane of the Reporting tool, select a report.
2. In the right pane, under the chart that you want to delete, click the Delete icon.

To display the trend report for CPU usage and memory usage for the last week

1. In the left pane of the Reporting tool, under Built-in Reports, expand System.
2. Click the report CPU vs. Memory Usage and HTTP Requests Rate.
3. In the right pane, on the report toolbar, click Duration, and then click Last Week.

To compare the bytes received rate and the bytes transmitted rate between two interfaces for the last week

1. In the right pane, on the report toolbar, click Create.
2. In the Report Name box, type a name for the custom report (for example, Custom_Interfaces), and then click OK. The report is created with the default System Overview chart, which displays the CPU Usage counter plotted for the last hour.
3. Under System Overview, on the chart toolbar, click Counters.
4. In the counter selection pane, in Title, type a name for the chart (for example, Interfaces bytes data).
5. In Plot chart for, click System entities statistics, and then in Select Group, select Interface.
6. On the Entities tab, click the interface name(s) you want to plot (for example, 1/1 and 1/2), and then click the > button.
7. On the Counters tab, click Bytes received (Rate) and Bytes transmitted (Rate) and then click the > button.

8. Click OK.
9. On the report toolbar, click Duration, and then click Last Week.

Updated: 2014-09-24

The data collection utility, nscollect, runs automatically when you start the NetScaler ADC. This utility retrieves the application performance data and stores it in the form of data sources on the ADC. You can create up to 32 data sources. The default data source is /var/log/db/default.

The data collection utility creates databases for global counters and entity-specific counters, and uses this data to generate reports. Global-counter databases are created at /var/log/db/<DataSourceName> . The entity-specific databases are created based on the entities configured on the NetScaler, and a separate folder is created for each entity type in /var/log/db/<DataSourceName/EntityNameDB>.

Nscollect retrieves data once every 5 minutes. It retains data in 5-minute granularity for one day, hourly for the last 30 days, and daily for three years.

You might have to stop and restart the data collection utility if data is not updated accurately or the reports display corrupted data.

To stop nscollect

At the command prompt, type:
/netscaler/nscollect stop

To start nscollect on the local system

At the command prompt, type:
/netscaler/nscollect start

AppFlow

May 29, 2015

The Citrix NetScaler appliance is a central point of control for all application traffic in the data center. It collects flow and user-session level information valuable for application performance monitoring, analytics, and business intelligence applications. It also collects web page performance data and database information. AppFlow transmits the information by using the Internet Protocol Flow Information eXport (IPFIX) format, which is an open Internet Engineering Task Force (IETF) standard defined in RFC 5101. IPFIX (the standardized version of Cisco's NetFlow) is widely used to monitor network flow information. AppFlow defines new Information Elements to represent application-level information, web page performance data, and database information.

Using UDP as the transport protocol, AppFlow transmits the collected data, called *flow records*, to one or more IPv4 collectors. The collectors aggregate the flow records and generate real-time or historical reports.

AppFlow provides visibility at the transaction level for HTTP, SSL, TCP, and SSL_TCP flows. You can sample and filter the flow types that you want to monitor.

AppFlow use actions and policies to send records for a selected flow to specific set of collectors. An AppFlow action specifies which set of collectors will receive the AppFlow records. Policies, which are based on Advanced expressions can be configured to select flows for which flow records will be sent to the collectors specified by the associated AppFlow action.

To limit the types of flows, you can enable AppFlow for a virtual server. AppFlow can also provide statistics for the virtual server.

You can also enable AppFlow for a specific service, representing an application server, and monitor the traffic to that application server.

Note: This feature is supported only on NetScaler nCore builds.
This topic includes the following details:

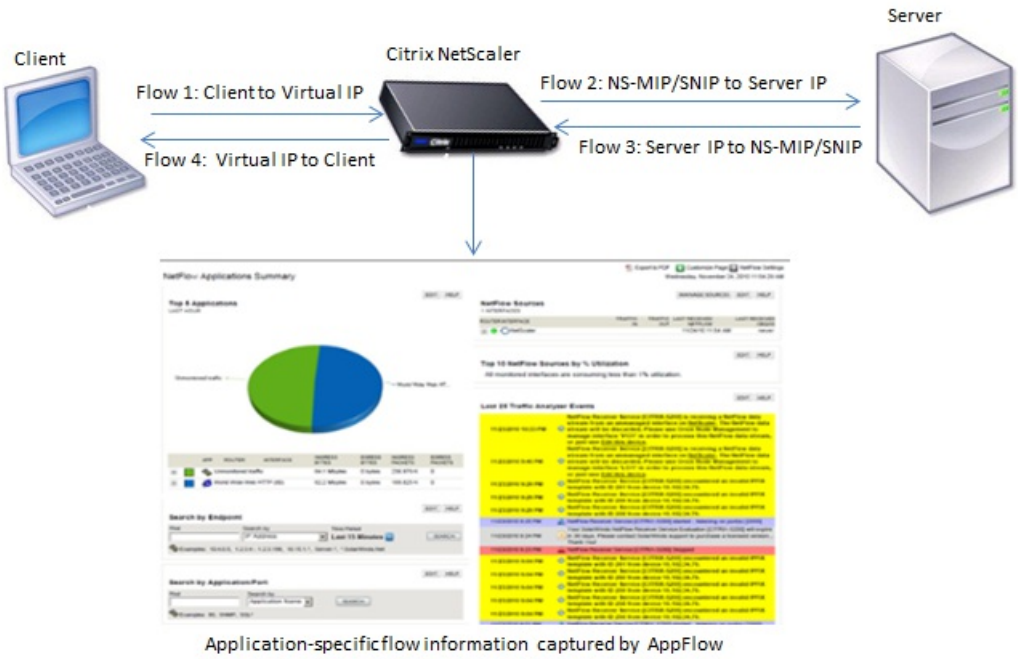
- How AppFlow Works
- Configuring the AppFlow Feature
- Exporting Performance Data of Web Pages to AppFlow Collector

Updated: 2015-05-28

In the most common deployment scenario, inbound traffic flows to a Virtual IP address (VIP) on the NetScaler appliance and is load balanced to a server. Outbound traffic flows from the server to a mapped or subnet IP address on the NetScaler and from the VIP to the client. A flow is a unidirectional collection of IP packets identified by the following five tuples: sourceIP, sourcePort, destIP, destPort, and protocol.

The following figure describes how the AppFlow feature works.

Figure 1. NetScaler Flow Sequence



Application-specific flow information captured by AppFlow

As shown in the figure, the network flow identifiers for each leg of a transaction depend on the direction of the traffic.

The different flows that form a flow record are:

Flow1: <Client-IP, Client-Port, VIP-IP, VIP-port, Protocol>

Flow2: <NS-MIP/SNIP, NS-port, Server-IP, Server-Port, Protocol>

Flow3: <Server-IP, Server-Port, NS-MIP/SNIP, NS-Port, Protocol>

Flow4: <VIP-IP, VIP-port, Client-IP, Client-Port, Protocol>

To help the collector link all four flows in a transaction, AppFlow adds a custom transactionID element to each flow. For application-level content switching, such as HTTP, it is possible for a single client TCP connection to be load balanced to different backend TCP connections for each request. AppFlow provides a set of records for each transaction.

This topic includes the following details:

- Flow Records
- Templates

Flow Records

Updated: 2013-08-20

AppFlow records contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. AppFlow records also contain application-level information (such as HTTP URLs, HTTP request methods and response status codes, server response time, and latency), web page performance data (such as page load time, page render time, and time spent on the page), and database information (such as database protocol, database response status and database response size). IPFIX flow records are based on templates that need to be sent before sending flow records.

Templates

AppFlow defines a set of templates, one for each type of flow. Each template contains a set of standard Information Elements (IEs) and Enterprise-specific Information Elements (EIEs). IPFIX templates define the order and sizes of the Information Elements (IE) in the flow record. The templates are sent to the collectors at regular intervals, as described in RFC 5101.

A template can include the following EIEs:

transactionID

An unsigned 32-bit number identifying an application-level transaction. For HTTP, this corresponds to a request and response pair. All flow records that correspond to this request and response pair have the same transaction ID. In the most common case, there are four uniflow records that correspond to this transaction. If the NetScaler generates the response by itself (served from the integrated cache or by a security policy), there may be only two flow records for this transaction.

connectionID

An unsigned 32-bit number identifying a layer-4 connection (TCP or UDP). The NetScaler flows are usually bidirectional, with two separate flow records for each direction of the flow. This information element can be used to link the two flows. For the NetScaler, connectionID is an identifier for the connection data structure to track the progress of a connection. In an HTTP transaction, for instance, a given connectionID may have multiple transactionID elements corresponding to multiple requests that were made on that connection.

tcpRTT

The round trip time, in milliseconds, as measured on the TCP connection. This can be used as a metric to determine the client or server latency on the network.

httpRequestMethod

An 8-bit number indicating the HTTP method used in the transaction. An options template with the number-to-method mapping is sent along with the template.

httpRequestSize

An unsigned 32-bit number indicating the request payload size.

httpRequestURL

The HTTP URL requested by the client.

httpUserAgent

The source of incoming requests to the Web server.

httpResponseStatus

An unsigned 32-bit number indicating the response status code.

httpResponseSize

An unsigned 32-bit number indicating the response size.

httpResponseTimeToFirstByte

An unsigned 32-bit number indicating the time taken to receive the first byte of the response.

httpResponseTimeToLastByte

An unsigned 32-bit number indicating the time taken to receive the last byte of the response.

flowFlags

An unsigned 64-bit flag used to indicate different flow conditions.

clientInteractionStartTime

Time at which the browser receives the first byte of the response to load any objects of the page such as images, scripts, and stylesheets.

clientInteractionEndTime

Time at which the browser received the last byte of response to load all the objects of the page such as images, scripts, and stylesheets.

clientRenderStartTime

Time at which the browser starts to render the page.

clientRenderEndTime

Time at which browser finished rendering the entire page, including the embedded objects.

dbProtocolName

An unsigned 8-bit number indicating the database protocol. Valid values are 1 for MS SQL and 2 for MySQL.

dbReqType

An unsigned 8-bit number indicating the database request method used in the transaction. For MS SQL, valid values are 1 is for QUERY, 2 is for TRANSACTION, and 3 is for RPC. For valid values for MySQL, see the MySQL documentation.

dbReqString

Indicates the database request string without the header.

dbRespStatus

An unsigned 64-bit number indicating the status of the database response received from the web server.

dbRespLength

An unsigned 64-bit number indicating the response size.

dbRespStatString

The response status string received from the web server.

Configuring the AppFlow Feature

May 28, 2015

You configure AppFlow in the same manner as most other policy-based features. First, you enable the AppFlow feature. Then you specify the collectors to which the flow records are sent. After that, you define actions, which are sets of configured collectors. Then you configure one or more policies and associate an action to each policy. The policy tells the NetScaler appliance to select requests the flow records of which are sent to the associated action. Finally, you bind each policy either globally or to specific vservers to put it into effect.

You can further set AppFlow parameters to specify the template refresh interval and to enable the exporting of httpURL, httpCookie, and httpReferer information. On each collector, you must specify the NetScaler IP address as the address of the exporter.
Note: For information about configuring the NetScaler as an exporter on the collector, see the documentation for the specific collector.
The configuration utility provides tools that help users define the policies and actions that determine exactly how the NetScaler appliance export records for a particular flow to a set of collectors(action.) The command line interface provides a corresponding set of CLI-based commands for experienced users who prefer a command line.

This topic includes the following details:

Updated: 2014-08-07

To be able to use the AppFlow feature, you must first enable it.

Note: AppFlow can be enabled only on nCore NetScaler appliances.

To enable the AppFlow feature by using the command line interface

At the command prompt, type one of the following commands:

enable ns feature AppFlow

To enable the AppFlow feature by using the configuration utility

Navigate to System > Settings, click Configure Advanced Features and select the AppFlow option.

Updated: 2014-08-07

A collector receives flow records generated by the NetScaler appliance. To be able to send flow records, you must specify

at least one collector. You can specify up to four. However, you cannot export the same data to multiple collectors. You can remove unused collectors. By default, the collector listens to IPFIX messages on UDP port 4739. You can change the default port, when configuring the collector. Similarly, by default, NSIP is used as the source IP for appflow traffic. You can change this default source IP to a SNIP or MIP address when configuring a collector.

To specify a collector by using the command line interface

At the command prompt, type the following commands to add a collector and verify the configuration:

- add appflow collector <name> -IPAddress <ipaddress> -port <port_number> -netprofile <netprofile_name>
- show appflow collector <name>

```
> add appflow collector col1 -IPaddress 10.102.29.251 -port 8000 -netprofile n2
```

To specify a collector by using the configuration utility

Navigate to System > AppFlow > Collectors, and create the AppFlow collector.

Updated: 2014-08-07

An AppFlow action is a set collectors, to which the flow records are sent if the associated AppFlow policy matches.

To configure an AppFlow action by using the command line interface

At the command prompt, type the following commands to configure an Appflow action and verify the configuration:

- add appflow action <name> --collectors <string> ... [-comment <string>]
- show appflow action

```
> add appflow action apfl-act-collector-1-and-3 -collectors collector-1 collecter-3
```

To configure an AppFlow action by using the configuration utility

Navigate to System > AppFlow > Actions, and create the AppFlow action.

Updated: 2014-08-07

After you configure an AppFlow action, you must next configure an AppFlow policy. An AppFlow policy is based on a rule, which consists of one or more expressions.

Note: For creating and managing AppFlow policies, the configuration utility provides assistance that is not available at the command line interface.

To configure an AppFlow policy by using the command line interface

At the command prompt, type the following command to add an AppFlow policy and verify the configuration:

- add appflow policy <name> <rule> <action>
- show appflow policy <name>

```
> add appflow policy apfl-pol-tcp-dsprt client.TCP.DSTPORT.EQ(22) apfl-act-collector-1-and-3
```

To configure an AppFlow policy by using the configuration utility

Navigate to System > AppFlow > Policies, and create the AppFlow policy.

To add an expression by using the Add Expression dialog box

1. In the Add Expression dialog box, in the first list box choose the first term for your expression.
   HTTP

   The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.

   SYS

   The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.

   CLIENT

   The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.

   When you make your choice, the rightmost list box lists appropriate terms for the next part of your expression.
2. In the second list box, choose the second term for your expression. The choices depend upon which choice you made in the previous step, and are appropriate to the context. After you make your second choice, the Help window below the Construct Expression window (which was blank) displays help describing the purpose and use of the term you just chose.
3. Continue choosing terms from the list boxes that appear to the right of the previous list box, or typing strings or numbers in the text boxes that appear to prompt you to enter a value, until your expression is finished.

Updated: 2014-08-07

To put a policy into effect, you must bind it either globally, so that it applies to all traffic that flows through the NetScaler, or to a specific virtual server, so that the policy applies only to the traffic related to that virtual server.

When you bind a policy, you assign it a priority. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer.

In the NetScaler operating system, policy priorities work in reverse order—the higher the number, the lower the priority. For example, if you have three policies with priorities of 10, 100, and 1000, the policy assigned a priority of 10 is performed first, then the policy assigned a priority of 100, and finally the policy assigned an order of 1000.

You can leave yourself plenty of room to add other policies in any order, and still set them to evaluate in the order you want, by setting priorities with intervals of 50 or 100 between each policy when you globally bind it. You can then add additional policies at any time without having to change the priority of an existing policy.

To globally bind an AppFlow policy by using the command line interface

At the command prompt, type the following command to globally bind an AppFlow policy and verify the configuration:

- bind appflow global <policyName> <priority> [<gotoPriorityExpression [-type <type>] [-invoke (<labelType> <labelName>)]
- show appflow global

bind appflow global af_policy_lb1_10.102.71.190 1 NEXT -type REQ_OVERRIDE -invoke vserver google

To bind an AppFlow policy to a specific virtual server by using the command line interface

At the command prompt, type the following command to bind an appflow policy to a specific virtual server and verify the configuration:
bind lb vserver <name> -policyname <policy_name> -priority <priority>

bind lb vserver google -policyname af_policy_google_10.102.19.179 -priority 251

To globally bind an AppFlow policy by using the configuration utility

Navigate to System > AppFlow, click AppFlow policy Manager and select the relevant Bind Point (Default Global) and Connection Type, and then bind the AppFlow policy.

To bind an AppFlow policy to a specific virtual server by using the configuration utility

Navigate to Traffic Management > Load Balancing > Virtual Servers, select the virtual server, and click Policies, and bind the AppFlow policy.

Updated: 2014-08-12

If you want to monitor only the traffic through certain virtual servers, enable AppFlow specifically for those virtual servers. You can enable AppFlow for load balancing, content switching, cache redirection, SSL VPN, GSLB, and authentication virtual servers.

To enable AppFlow for a virtual server by using the command line interface

At the command prompt, type:

set cs vserver <name> <protocol> <IPAddress> <port> -appflowLog ENABLED

> set cs vserver Vserver-CS-1 HTTP 10.102.29.161 80 -appflowLog ENABLED

To enable AppFlow for a virtual server by using the configuration utility

Navigate to Traffic Management > Content Switching > Virtual Servers, select the virtual server, and enable AppFlow Logging option.

Updated: 2014-08-07

You can enable AppFlow for services that are to be bound to the load balancing virtual servers.

To enable AppFlow for a service by using the command line interface

At the command prompt, type:

set service <name> -appflowLog ENABLED

```
set service ser -appflowLog ENABLED
```

To enable AppFlow for a service by using the configuration utility

Navigate to Traffic Management > Load Balancing > Services, select the service, and enable AppFlow Logging option.

Updated: 2014-08-08

You can set AppFlow parameters to customize the exporting of data to the collectors.

To set the AppFlow Parameters by using the command line interface

At the command prompt, type the following commands to set the AppFlow parameters and verify the settings:

- set appflow param [-templateRefresh <secs>] [-appnameRefresh <secs>] [-flowRecordInterval <secs>] [-udpPmtu <positive_integer>] [-httpUrl (**ENABLED** | **DISABLED**)] [-httpCookie (**ENABLED** | **DISABLED**)] [-httpReferer (**ENABLED** | **DISABLED**)] [-httpMethod (**ENABLED** | **DISABLED**)] [-httpHost (**ENABLED** | **DISABLED**)] [-httpUserAgent (**ENABLED** | **DISABLED**)] [-httpXForwardedFor (**ENABLED** | **DISABLED**)][-clientTrafficOnly (**YES** | **NO**)]
- show appflow Param

```
> set appflow Param -templateRefresh 240 -udpPmtu 128 -httpUrl enabled
```

To set the AppFlow parameters by using the configuration utility

Navigate to System > AppFlow, click Change AppFlow Settings, and specify relevant AppFlow parameters.

Updated: 2013-08-20

The following example illustrates the procedure for configuring AppFlow for DataStream using the command line interface.

```
> enable feature appflow
> add db user sa password freebsd
> add lbvserver lb0 MSSQL 10.102.147.97 1433 -appflowLog ENABLED
> add service sv0 10.103.24.132 MSSQL 1433 -appflowLog ENABLED
> bind lbvserver lb0 sv0
> add appflow collector col0 -IPAddress 10.102.147.90
> add appflow action act0 -collectors col0
> add appflow policy pol0 "mssql.req.query.text.contains(\"select\")" act0
> bind lbvserver lb0 -policyName pol0 -priority 10
```

When the Netscaler appliance receives a database request, the appliance evaluates the request against a configured policy. If a match is found, the details are sent to the AppFlow collector configured in the policy.

Exporting Performance Data of Web Pages to AppFlow Collector

May 28, 2015

The EdgeSight Monitoring application provides web page monitoring data with which you can monitor the performance of various Web applications served in a Netscaler environment. You can now export this data to AppFlow collectors to get an in-depth analysis of the web page applications. AppFlow, which is based on IPFIX standard, provides more specific information about web application performance than does EdgeSight monitoring alone.

You can configure both load balancing and content switching virtual servers to export EdgeSight Monitoring data to AppFlow collectors. Before configuring a virtual server for AppFlow export, associate an Appflow action with the EdgeSight Monitoring responder policy.

The following web page performance data is exported to AppFlow:

- **Page Load Time**. Elapsed time, in milliseconds, from when the browser starts to receive the first byte of a response until the user starts to interact with the page. At this stage, all the page content might not be loaded.
- **Page Render Time**.Elapsed time, in milliseconds, from when the browser receives the first byte of response until either all page content has been rendered or the page load action has timed out.
- **Time Spent on the Page**. Time spent by users on a page. Represents the period of time from one page request to the next one.

AppFlow transmits the performance data by using the Internet Protocol Flow Information eXport (IPFIX) format, which is an open Internet Engineering Task Force (IETF) standard defined in RFC 5101. The AppFlow templates use the following enterprise-specific Information Elements (EIEs) to export the information:

- **Client Load End Time**. Time at which the browser received the last byte of a response to load all the objects of the page such as images, scripts, and stylesheets.
- **Client Load Start Time**.Time at which the browser receives the first byte of the response to load any objects of the page such as images, scripts, and stylesheets.
- **Client Render End Time**. Time at which browser finished rendering the entire page, including the embedded objects.
- **Client Render Start Time**. Time at which the browser started rendering the page.

This topic includes the following details:

- Prerequisites for Exporting Performance Data of Web Pages to AppFlow Collectors
- Associating an AppFlow Action with the EdgeSight Monitoring Responder Policy

Updated: 2013-09-13

Before associating the AppFlow action with the AppFlow policy, verify that the following prerequisites have been met:

- The AppFlow feature has been enabled and configured. For instructions, see "Configuring the AppFlow feature".
- The Responder feature has been enabled. For instructions, see "Enabling a Responder Feature".
- The EdgeSight Monitoring feature has been enabled. For instructions, see "Enabling an Application for EdgeSight Monitoring."
- EdgeSight Monitoring has been enabled on the load balancing or content switching virtual servers bound to the services of applications for which you want to collect the performance data. For instructions, see "Enabling an Application for

EdgeSight Monitoring."

Updated: 2013-10-31

To export the web page performance data to the AppFlow collector, you must associate an AppFlow action with the EdgeSight Monitoring responder policy. An AppFlow action specifies which set of collectors receive the traffic.

To associate an AppFlow action with the EdgeSight Monitoring Responder policy by using the command line interface

At the command prompt, type:

set responder policy <name> -appflowAction <action_Name>

```
set responder policy pol -appflowAction actn
```

To associate an AppFlow action with the EdgeSight Monitoring Responder policy by using the configuration utility

1. Navigate to AppExpert > Responder > Policies.
2. In the details pane, select an EdgeSight Monitoring responder policy, and then click **Open**.
3. In the **Configure Responder Policy** dialog box, in the **AppFlow Action** drop-down list, select the AppFlow action associated with the collectors to which you want to send the web-page performance data.
4. Click **OK**.

Configuring a Virtual Server to Export EdgeSight Statistics to Appflow Collectors

To export EdgeSight statistics information from a virtual server to the AppFlow collector, you must associate an AppFlow action with the virtual server.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers or Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select a virtual server, or multiple virtual servers, and then click Enable EdgeSight Monitoring.
3. In the Enable EdgeSight Monitoring dialog box, select the Export EdgeSight statistics to Appflow check box.
4. From the Appflow Action drop-down list, select the AppFlow action. The AppFlow action defines the list of AppFlow collectors to which it exports EdgeSight Monitoring statistics. If you have selected multiple load balancing virtual servers, the same AppFlow Action will be associated with the responder policies bound to them. You can later change the AppFlow Action configured for each of the selected Load Balancing virtual server individually, if required.
5. Click OK.

AutoScale

Aug 26, 2013

Efficient hosting of applications in a cloud requires continuous optimization of application availability. To meet increasing demand, you have to scale network resources upward. When demand subsides, you need to scale down to avoid the unnecessary cost of idle resources. To minimize the cost of running the application by deploying only as many instances as are necessary during any given period of time, you have to constantly monitor traffic. However, monitoring traffic manually is not a feasible option. For the application environment to be able to scale up or down rapidly, you need to automate the processes of monitoring traffic and of scaling resources up and down whenever necessary.

If your organization uses Citrix CloudPlatform to deploy and manage the cloud environment, a Citrix NetScaler appliance can automatically scale users' applications as needed. The CloudPlatform elastic load balancing feature includes a feature called *AutoScale*. A CloudPlatform user can use the AutoScale feature to specify thresholds for various conditions for automatically scaling the application fleet upward and downward. The scale-up and scale-down conditions can vary from simple use cases, such as a server's CPU usage, to complex use cases, such as a combination of a server's CPU usage and responsiveness. CloudPlatform, in turn, configures the NetScaler appliance to load balance traffic to the application virtual machines (VMs), monitor application thresholds and performance, and trigger scale-up and scale-down actions to add or remove VMs to or from the application fleet.

The CloudPlatform user performs all AutoScale configuration tasks by using the CloudPlatform user interface or APIs. The CloudPlatform user:

1. Creates a load balancing rule, with the necessary load balancing algorithm and stickiness.
2. Configures AutoScale parameters by specifying the application instance template, the minimum number of instances to maintain, the maximum number of instances permitted, scale-up and scale-down policies, and other information necessary for the functioning of the feature.
3. Submits the configuration.

For information about configuring a load balancing rule and AutoScale, see *Citrix CloudPlatform 3.0.5 (powered by Apache CloudStack) Administrator's Guide*, at http://support.citrix.com/article/CTX134823.

When the CloudPlatform user completes the AutoScale configuration, CloudPlatform uses the NetScaler NITRO API to push all the necessary configuration commands to the NetScaler appliance. As the NetScaler administrator, you do not have to perform any tasks for configuring AutoScale on the NetScaler appliance. However, you might have to be aware of certain prerequisites, and you might have to troubleshoot the configuration if issues arise in the AutoScale configuration. To troubleshoot the configuration, you have to be aware of how CloudPlatform works and what configuration CloudPlatform pushes to the NetScaler appliance. You also need a working knowledge of how to troubleshoot issues on a NetScaler appliance.

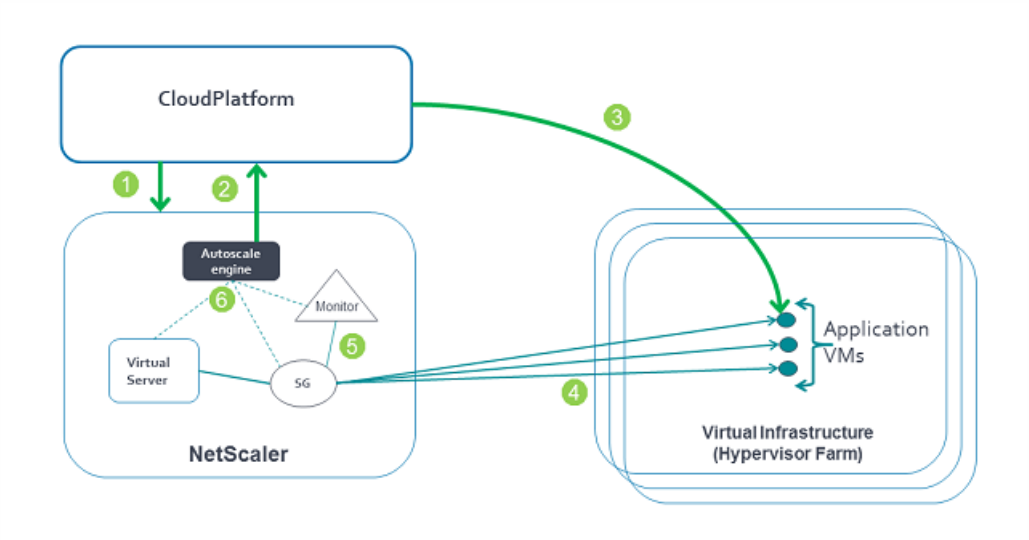
How AutoScale Works

Sep 06, 2013

When the CloudPlatform user completes the AutoScale configuration, CloudPlatform uses the NetScaler NITRO API to create an AutoScale-related configuration on the NetScaler appliance. For information about the configuration commands that CloudPlatform uses to configure the NetScaler appliance, see "NetScaler Configuration Details."

The following diagram shows the sequence of operations, beginning with CloudPlatform pushing the AutoScale configuration to the NetScaler appliance. The events are numbered in the order in which they occur, and are described below.

Figure 1. AutoScale Architecture



When the CloudPlatform user submits the AutoScale configuration, the following events occur:

1. CloudPlatform uses the NetScaler NITRO API to push the AutoScale configuration to the NetScaler appliance, creating AutoScale-related entities on the appliance. The entities include a load balancing virtual server, a service group, and monitors.
2. The AutoScale engine on the NetScaler appliance sends API requests to CloudPlatform to initially deploy the minimum number of virtual machines required.
3. CloudPlatform provisions the minimum number of instances (VMs) on the hypervisors (virtualization hosts) that it manages.
4. The NetScaler appliance discovers the IP addresses assigned by CloudPlatform to the newly created VMs and binds them, as services, to the service group representing them. The NetScaler appliance can then load balance traffic to the VMs.
5. NetScaler monitors bound to the service group start monitoring the load by collecting SNMP metrics from the instances.
6. The AutoScale engine on the NetScaler appliance monitors the metrics collected from the VMs and triggers scale-up and scale-down events whenever the metrics breach the configured threshold for the specified period. As part of the scale-up trigger, the NetScaler AutoScale engine sends an API request to CloudPlatform to deploy a new VM. After the virtual machine is deployed, the AutoScale engine binds the service representing the VM (IP address and port) to the service group and, after the configured quiet time, starts forwarding load balanced traffic to the new virtual machine. Likewise, as part of the scale-down trigger, the NetScaler AutoScale engine selects a VM, stops forwarding new requests to that

instance, and waits for the configured quiet time (to allow for the processing of current requests to complete) before it sends an API request to CloudPlatform to destroy the chosen instance.

In this way, the NetScaler appliance monitors the application and triggers scale-up and scale-down events on the basis of application load and/or performance.

Supported Environment

Sep 30, 2013

AutoScale is supported in the following environment:

- Citrix CloudPlatform 3.0.5.
- Citrix NetScaler MPX/SDX/virtual appliance running Citrix NetScaler release 10.e and later.
- SNMP v1/v2.

Prerequisites

Sep 06, 2013

Before you set up AutoScale, do the following:

- Make sure that CloudPlatform is reachable from the NetScaler appliance. You can do so by logging on to the NetScaler appliance and sending ping requests to the CloudPlatform server's IP address.
- Make sure that the network service offering used in CloudPlatform includes the NetScaler appliance as an external load balancing device.
- Use a CloudPlatform and NetScaler release that supports AutoScale. For information about NetScaler releases that support AutoScale, see "Supported Environment."

If you have to troubleshoot an AutoScale setup, you also have to know the prerequisites for setting up AutoScale in CloudPlatform. See the "Prerequisites" section of "Configuring AutoScale" in the *Citrix CloudPlatform 3.0.5 (powered by Apache CloudStack) Administrator's Guide*, at http://support.citrix.com/article/CTX134823.

NetScaler Configuration Details

Mar 19, 2012

The following table describes the AutoScale configuration commands that are used by Citrix CloudPlatform to configure a NetScaler appliance.

Table 1. NetScaler Configuration for AutoScale

AutoScale configuration command(s)	Description
add lb vserver Cloud-VirtualServer-192.0.2.116-22 TCP 192.0.2.116 22 -persistenceType NONE -lbMethod ROUNDROBIN -cltTimeout 9000 -minAutoscaleMembers 2 -maxAutoscaleMembers 5	Creates a load balancing virtual server to evenly distribute the load on the application instances (ROUND-ROBIN method). The virtual server also specifies the limits for the number of instances to which the application can scale up or down (maxAutoscaleMembers and minAutoscaleMembers, respectively).
add serviceGroup Clouda35a6b6b76614006b97476e841b80f79 TCP -maxClient 0 -maxReq 0 -cip DISABLED -usip NO -useproxyport YES -cltTimeout 9000 -svrTimeout 9000 -CKA NO -TCPB NO -CMP NO -autoScale POLICY -memberPort 22 bind lb vserver Cloud-VirtualServer-192.0.2.116-22 Clouda35a6b6b76614006b97476e841b80f79	Creates an AutoScale service group for the application instances, with the service group's autoScale parameter set to POLICY. Also specifies the port on which the service group members must receive traffic. The second command binds the service group to the load balancing virtual server.
add server autoscale-internal_server_Clouda35a6b6b76614006b97476e841b80f79 autoscale-internal_server_Clouda35a6b6b76614006b97476e841b80f79 bind serviceGroup Clouda35a6b6b76614006b97476e841b80f79 autoscale-internal_server_Clouda35a6b6b76614006b97476e841b80f79 22	Creates a server entry to represent the application instances. Binds the server entry to the service group.
add lb metricTable Cloud-MTbl-192.0.2.116-22 bind lb metricTable Cloud-MTbl-192.0.2.116-22 Linux_User_CPU_-_percentage 1.3.6.1.4.1.2021.11.9.0 add lb monitor Cloud-Mon-192.0.2.116-22 LOAD -interval 24 -destPort 161 -snmpCommunity public -metricTable Cloud-MTbl-192.0.2.116-22	Configures a new SNMP monitor to retrieve the specified metrics.

AutoScale configuration command(s)	Description
bind lb monitor Cloud-Mon-192.0.2.116-22 -metric Linux_User_CPU_-_percentage -metricThreshold 2147483647 bind serviceGroup Clouda35a6b6b76614006b97476e841b80f79 -monitorName Cloud-Mon-192.0.2.116-22 -passive	
add autoscale profile Cloud-AutoScale-Profile-192.0.2.116-22 -type CLOUDSTACK -url "http://10.102.31.107:8080/client/api" -apiKey t0fEWPtk_ncQYbofjAm1jjlgGTR7UNZrkZ3sdEpLREBNzBPLSNpNz8qNSbc439xNtYnEYdWn_MsUC_CUazaIKg -sharedSecret -PrE5h3DP7swHAN12TGBlX-xSTRLHzob91l6O0VO1FMxvE1UOl7uoD6_Z0bkkLaVtK5Y10oBkTzgbTwp3u5lCQ	Creates an AutoScale profile to specify the details required by NetScaler for making API requests to CloudPlatform (URL, API key, and shared secret).
add autoscale action Cloud-AutoScale-ScaleUpAction-192.0.2.116-22 -type SCALE_UP -profileName Cloud-AutoScale-Profile-192.0.2.116-22 -parameters "command=deployVirtualMachine&zoneid=2ab23590-78cb-4106-8d85-4412a2f2435f&serviceofferingid=b9503e47-0d8f-4c89-a88d-04d8b17fe8e9&templateid=1a4a5084-208c-47a8-9c16-d582550cf759&displayname=AutoScale-LB-lb&networkids=a3c97129-b729-4c72-994f-7b918f20ce4d&lbruleid=f96b7f3b-19ec-4123-891c-604f05b032b3" -quietTime 90 -vServer Cloud-VirtualServer-192.0.2.116-22	Creates a scale-up action, which enables the NetScaler appliance to add virtual machines (instances) to the application fleet.
add autoscale action Cloud-AutoScale-ScaleDownAction-192.0.2.116-22 -type SCALE_DOWN -profileName Cloud-AutoScale-Profile-192.0.2.116-22 -parameters "command=destroyVirtualMachine&lbruleid=f96b7f3b-19ec-4123-891c-604f05b032b3" -vmDestroyGracePeriod 30 -quietTime 90 -vServer Cloud-VirtualServer-192.0.2.116-22	Creates a scale-down action, which enables the NetScaler appliance to remove virtual machines (instances) from the application fleet.
add autoscale policy Cloud-AutoScale-Policy-Min-192.0.2.116-22 -rule "SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").ACTIVESERVICES.LT(SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").MINAUTOSCALEMEMBERS)" -action Cloud-AutoScale-ScaleUpAction-192.0.2.116-22	Creates an AutoScale policy to initially create the specified minimum number of VMs and, later, to ensure that the number of VMs in the fleet does not fall below the required minimum.
add autoscale policy Cloud-AutoScale-Policy-Max-192.0.2.116-22 -rule "SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").ACTIVESERVICES.GT(SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").MAXAUTOSCALEMEMBERS)" -action Cloud-AutoScale-ScaleDownAction-192.0.2.116-22	Creates an AutoScale policy to prevent the number of VMs in the fleet from exceeding the specified maximum.
add autoscale policy Cloud-AutoScale-Policy-192.0.2.116-22-35 -rule "SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").ACTIVESERVICES.LT(SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").MAXAUTOSCALEMEMBERS) && (SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").SNMP_TABLE(0).AVERAGE_VALUE.GT(90))" -action Cloud-AutoScale-ScaleUpAction-192.0.2.116-22	Creates an AutoScale policy to evaluate the metrics that are collected and trigger a scale-up action when the metric value breaches the threshold specified for the scale-up policy.

AutoScale configuration command(s)	Description
add autoscale policy Cloud-AutoScale-Policy-192.0.2.116-22-36 -rule "SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").ACTIVESERVICES.GT(SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").MINAUTOSCALEMEMBERS) && (SYS.VSERVER(\"Cloud-VirtualServer-192.0.2.116-22\").SNMP_TABLE(0).AVERAGE_VALUE.LT(30))" -action Cloud-AutoScale-ScaleDownAction-192.0.2.116-22	Creates an AutoScale policy to evaluate the collected metrics and trigger a scale-down action when the metric value breaches the threshold specified by the scale-down policy.
add ns timer Cloud-AutoScale-Timer-192.0.2.116-22 -interval 30 bind ns timer Cloud-AutoScale-Timer-192.0.2.116-22 -policyName Cloud-AutoScale-Policy-Min-192.0.2.116-22 -priority 1 -gotoPriorityExpression END -sampleSize 1 -threshold 1 bind ns timer Cloud-AutoScale-Timer-192.0.2.116-22 -policyName Cloud-AutoScale-Policy-Max-192.0.2.116-22 -priority 2 -gotoPriorityExpression END -sampleSize 1 -threshold 1 bind ns timer Cloud-AutoScale-Timer-192.0.2.116-22 -policyName Cloud-AutoScale-Policy-192.0.2.116-22-35 -priority 3 -gotoPriorityExpression END -sampleSize 2 -threshold 2 bind ns timer Cloud-AutoScale-Timer-192.0.2.116-22 -policyName Cloud-AutoScale-Policy-192.0.2.116-22-36 -priority 4 -gotoPriorityExpression END -sampleSize 2 -threshold 2	Creates a timer that enables evaluation of the AutoScale policies at the configured sampling intervals.

Troubleshooting

Sep 06, 2013

Before you attempt to resolve an AutoScale issue, make sure that the prerequisites have been adhered to, on both the CloudPlatform server and the NetScaler appliance, as described in "Prerequisites." If that does not resolve the issue, your problem could be one of the following.

- Recommend that the CloudPlatform user deploy one VM manually in the network before configuring AutoScale. Ask the user to remove the AutoScale configuration from the NetScaler appliance or the load balancer from the network, manually deploy one VM (preferably using the template created for the AutoScale configuration), and then create the AutoScale configuration.
- Verify that the CloudPlatform user has configured the VM template in such a way that the VMs that are created from the template can accept traffic without manual intervention. If a provisioned VM cannot accept traffic automatically, the metric remains above the threshold, and the AutoScale configuration continues to provision additional VMs, as designed. To remedy the issue, disable AutoScale from CloudPlatform, fix the template, and then enable AutoScale.
- Verify that the CloudPlatform user has not exceeded the limit for the number of VMs imposed by the user's account.
- Verify that the CloudPlatform server is up and is reachable from the NetScaler appliance.
- Verify that the CloudPlatform log file, management-server.log, has reported the successful creation of the AutoScale configuration in CloudPlatform.
- Verify that the scale-up policy that is responsible for initial scale up (the policy name is prefixed with Cloud-AutoScale-Policy-Min) is receiving hits.

- Verify that the CloudPlatform user has configured the VM template in such a way that the VMs that are created from the template can accept traffic without manual intervention. If a provisioned VM cannot accept traffic automatically, the metric remains above the threshold, and the AutoScale configuration continues to provision additional VMs, as designed. To remedy the issue, disable AutoScale from CloudPlatform, fix the template, and then enable AutoScale.
- Verify that the quiet time that the CloudPlatform user has configured in the AutoScale configuration is sufficient to ensure even traffic distribution to all the VMs, including the new VM. If the quiet time is too low, and traffic distribution has not stabilized, the metrics might remain above the threshold, and additional VMs might be spawned.

- Verify that the CloudPlatform user has installed an SNMP agent in the VM template, and that the SNMP agent is up and running on every VM.
- Verify that the CloudPlatform user has not exceeded the limit for the number of VMs imposed by the user's account.
- Verify that the CloudPlatform user has correctly configured the SNMP parameters to collect metrics from the VM (for example, the community string and the port).
- Verify that the scale-up or scale-down policy is receiving hits.
- Verify that the CloudPlatform server is up, and that the CloudPlatform server is reachable from the NetScaler appliance.

- Verify that the user has configured the templates in such a way that the VMs created from the templates can start serving traffic without any manual intervention.
- Verify that the service is running on the VMs, on the configured member port.
- Send a ping request to the gateway (virtual router), from the VM that is not accepting traffic.

- The VMs might not be deleted immediately after the AutoScale configuration is deleted. Wait for about 5 minutes after you have deleted the AutoScale configuration, and then check again.
- If the destruction of VMs has not commenced after 5 minutes, you might have to delete the VMs manually.

Clustering

May 25, 2015

A NetScaler cluster is a group of nCore appliances working together as a single system image. Each appliance of the cluster is called a node. The cluster can include as few as 2 or as many as 32 NetScaler nCore hardware or virtual appliances as nodes.

The client traffic is distributed between the nodes to provide high availability, high throughput, and scalability.

To create a cluster, you must add the appliances as cluster nodes, set up communication between the nodes, set up links to the client and server networks, configure the appliances, and configure the distribution of client and server traffic.

NetScaler Feature-level Support in a Cluster

Jun 15, 2015

The following table lists the features that are not supported in the NetScaler 10 release and also states the support for the feature in subsequent NetScaler releases.

Note:

- Unless stated otherwise in this table, all NetScaler features are supported in a cluster.
- The entry "Node-level" in the table indicates that the feature is supported only on individual cluster nodes.

NetScaler Feature	10	10.1	10.5	11
SSL (classic policies) Note: SSL - advanced policies are supported from NetScaler 10 onwards.	No	No	No	No
SSL FIPS	No	No	No	No
SSL Certificate Bundle	No	No	No	No
Content switching actions Note: The content switching feature is supported from NetScaler 10 onwards.	No	Yes	Yes	Yes
Policy-based logging for content switching policies	No	Yes	Yes	Yes
Rate limiting	No	Yes	Yes	Yes
Action analytics	No	Yes	Yes	Yes
Branch Repeater load balancing	No	Yes	Yes	Yes
GSLB	No	No	Yes	Yes
RTSP	No	No	No	No
DNSSEC	No	No	No	No
DNS64	No	No	No	No
FTP	No	No	No	Yes

NetScaler Feature	10	10.1	10.5	11
TFTP	No	No	No	No
Connection mirroring	No	No	No	No
Integrated caching	Node-Level	Node-level	Node-level	Node-level
Large shared cache	No	Node-level	Node-level	Node-level
Application firewall	No	No	Node-level	Node-level
HTTP Denial-of-Service Protection (HDOSP)	Node-level	Node-level	Node-level	Node-level
Priority queuing (PQ)	Node-level	Node-level	Node-level	Node-level
Sure connect (SC)	Node-level	Node-level	Node-level	Node-level
AppQoE	NA	Node-level	Yes	Yes
Surge protection	Node-level	Node-level	Node-level	Node-level
MPTCP	No	No	Yes	Yes
MSR	Yes	Yes	Yes	Yes Note: Not supported in a L3 cluster.
IS-IS (IPv4 and IPv6)	No	Yes	Yes	Yes
IP-IP tunneling	No	Yes	Yes	Yes
Link load balancing	No	No	Yes	Yes
FIS (Failover Interface Set)	No	No	Yes	Yes
Link redundancy (LR)	No	No	Yes	Yes
NAT46	No	No	No	No

NetScaler Feature	10	10.1	10.5	11
NAT64	No	No	No	No
v6 ReadyLogo	No	No	No	No
Traffic domains	No	No	Yes	Yes
Route monitor	No	No	No	No
GRE tunneling (CB)	No	No	No	No
Layer 2 mode	No	No	Yes	Yes
Net profiles	No	No	Yes	Yes
HTTPS callout	No	Yes	Yes	Yes
AAA-TM	No	Node-level	Node-level	Node-level
AppFlow	No	Node-level	Node-level	Node-level
Insight	No	No	No	No
HDX Insight	No	No	No	No
VMAC/VRRP	No	No	Yes	Yes
NetScaler Push	No	No	No	No
Stateful Connection Failover	No	No	No	No
Graceful Shutdown	No	No	No	No
DBS AutoScale	No	No	No	No
DSR using TOS	No	No	No	No
Finer Startup-RR Control	Node-level	Node-level	Node-level	Node-level

NetScaler Feature	10	10.1	10.5	11
XML XSM	No	No	No	No
DHCP RA	No	No	No	No
Bridge Group	No	No	Yes Note: Supported from NetScaler 10.5 Build 52.1115.e onwards.	Yes
Network Bridge	No	No	No	No
Web Interface on NetScaler (WIonNS)	No	No	No	No
EdgeSight Monitoring	No	No	No	No
Metrics tables - Local	No	No	No	No
DNS Caching	Node-level	Node-level	Node-level	Node-level
Call Home	Node-level	Node-level	Node-level	Node-level
NetScaler Gateway or SSL VPN	No	No	Node-level	Node-level
CloudBridge Connector	No	No	No	No

Prerequisites for Cluster Nodes

Jan 28, 2016

NetScaler appliances that are to be added to a cluster must satisfy the following criteria:

- A NetScaler cluster can only include NetScaler nCore appliances. Clustering of NetScaler Classic appliances is not supported.
- All appliances must have the same software version and build.
- All appliances must be of the same platform type. This means that a cluster must have either all hardware appliances (MPX) or virtual appliances (VPX) or SDX NetScaler instances.

  Note:
  - For a cluster of hardware appliances (MPX), all appliances must be of the same model type.
  - For a cluster of virtual appliances (VPX), the appliances must be deployed on the following hypervisors: XenServer, Hyper-V, VMware ESX, and KVM.
  - Clustering of SDX NetScaler instances is supported in NetScaler 10.1 and later releases. To create a cluster of SDX NetScaler instances, see "Setting up a Cluster of NetScaler Instances".
- All appliances must be on the same network.
- All appliances must have the same licenses. Also, depending on the NetScaler version, there are some additional aspects to address:
  - For releases prior to NetScaler 10.5 Build 52.x:
    - A separate cluster license file is required. This file must be copied to the /nsconfig/license/ directory of the configuration coordinator.
    - Because of the separate cluster license file, the cluster feature is available irrespective of the NetScaler license.
  - For releases after NetScaler 10.5 Build 52.x:
    - No separate cluster license is required.
    - Cluster is licensed with the Enterprise and Platinum licenses. Cluster is not available for Standard license.
- Be initially configured and connected to a common client-side and server-side network.

Note: For a cluster of virtual appliances, that has large configurations, it is recommended to use 6 GB RAM for each node of the cluster.

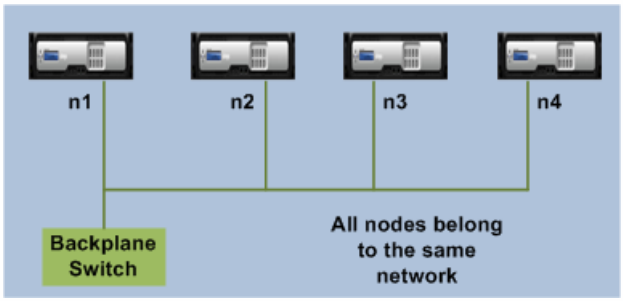
Cluster Overview

Jun 15, 2015

A NetScaler cluster is formed by grouping NetScaler appliances together. Based on the network location of the NetScaler appliances that you intend adding to the cluster, you must be aware of the following cluster setups:

Note: Unless specified otherwise, cluster features and configurations are the same for L2 and L3 clusters.

- **L2 cluster:** In this cluster deployment, all cluster nodes belong to the same network.



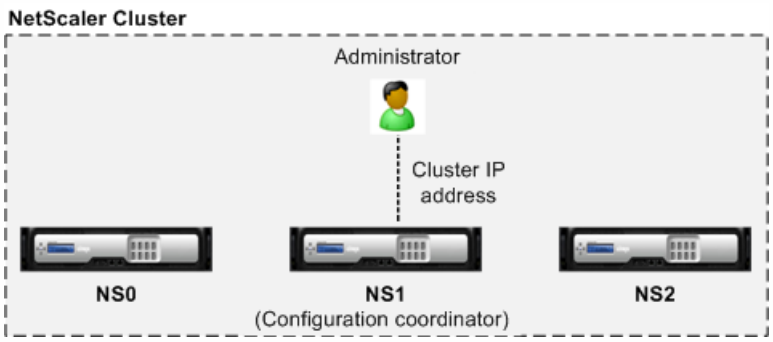
- **L3 cluster:** In this cluster deployment, cluster nodes can belong to different networks.

  Note: Not yet supported. Likely to be supported in future NetScaler versions.

Synchronization Across Cluster Nodes

May 25, 2015

All configurations on a NetScaler cluster are performed on the cluster IP address, which is the management address of the cluster. This cluster IP address is owned by a cluster node that is referred to as the cluster configuration coordinator as shown in the following figure:



The configurations that are available on the configuration coordinator are automatically propagated to the other cluster nodes and therefore all cluster nodes have the same configurations.

Note:

- NetScaler allows only a few configurations to be performed on individual cluster nodes through their NetScaler IP (NSIP) address. These configurations are not propagated across the other cluster nodes. For more information, see "Operations Supported on Individual Cluster Nodes".
- The following commands when executed on the cluster IP address are not propagated to other cluster nodes:
  - shutdown: Shuts down only the configuration coordinator.
  - reboot: Reboots only the configuration coordinator.
  - rm cluster instance: Removes the cluster instance from the node that you are executing the command on.
- If the NetScaler cluster is configured to use a quorum (quorum is mandatory for versions prior to NetScaler 10.5), a command is propagated to the other cluster nodes only when a majority of the nodes are in synch. If a majority of the nodes are not in synch or are in the process of synchronizing, the new commands cannot be accepted and therefore command propagation is temporarily halted.

When a node is added to a cluster, the configurations and the files (SSL certificates, licenses, DNS, and so on) that are available on the cluster configuration coordinator are synchronized to the newly added cluster node. When an existing cluster node, that was intentionally disabled or that had failed, is once again added, the cluster compares the configurations available on the node with the configurations available on the configuration coordinator. If there is a mismatch in configurations, the node is synchronized by using one of the following:

- **Full synchronization.** If the difference between configurations exceeds 255 commands, all the configurations of the configuration coordinator are applied to the node that is rejoining the cluster. The node remains operationally unavailable for the duration of the synchronization.
- **Incremental Synchronization.** If the difference between configurations is less than or equal to 255 commands, only the configurations that are not available are applied to the node that is rejoining the cluster. The operational state of the node remains unaffected.

Note: You can also manually synchronize the configurations and files. For more information, see "Synchronizing Cluster

Configurations" and "Synchronizing Cluster Files".

Striped, Partially Striped, and Spotted Configurations

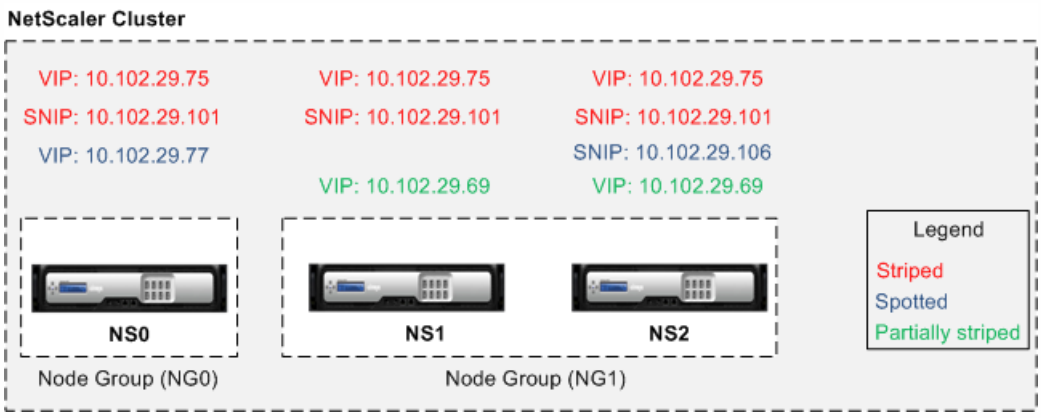
May 25, 2015

By virtue of command propagation, all nodes in a cluster have the same configurations. However, you may want some configurations to be available only on certain cluster nodes. While you cannot restrict the nodes on which the configurations are available, you can specify the nodes on which the configurations are active.

For example, you can define a SNIP address to be active on only one node, or define a SNIP address to be active on all nodes, or define a VIP address to be active on only one node, or define a VIP address to be active on all nodes, or define a VIP address to be active only on two nodes of a 3-node cluster.

Depending on the number of nodes the configurations are active on, cluster configurations are referred to as striped, partially striped, or spotted configurations.

Figure 1. Three-node cluster with striped, partially striped, and spotted configurations



The following table provides more details on the types of configurations:

Configuration Type	Active on...	Applicable to...	Configurations...
Striped configuration	All the cluster nodes	All entities	No specific configuration required to make an entity striped. By default, all entities defined on a cluster IP address are striped on all the cluster nodes.
Partially striped configuration	A subset of cluster nodes	Refer "Node Groups"	Bind the entities that you want to be partially striped, to a node group. The configuration will be active only on the cluster nodes that belong to the node group.
Spotted configuration	Single cluster node	• SNIP address • SNMP Engine ID • Hostname of cluster	A spotted configuration can be defined using one of two approaches. • **SNIP address.** When creating the SNIP address, specify the node on which you want the SNIP address to be active, as the owner node. **Example:**

Configuration Type	Active on...	Applicable to...	Configurations...
		nodes • Entities that can be bound to a node group	add ns ip 10.102.29.106 255.255.255.0 -type SNIP -ownerNode 2 (assuming node NS2 ID is 2) Note: You cannot change the ownership of a spotted SNIP address at run time. To change the ownership, you must first delete the SNIP address and add it again by specifying the new owner. • **Entities that can be bound to a node group.** By binding the entity to a single-member node group.

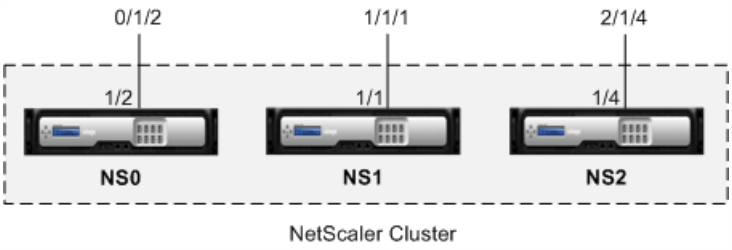
Note: Citrix recommends that you use spotted SNIP addresses. You can use striped SNIP addresses only if there is a shortage of IP addresses. The use of striped IP addresses can result in ARP flux issues.

Communication in a Cluster Setup

May 25, 2015

The interfaces of NetScaler appliances that are added to a cluster, are prefixed with a node ID. This helps identify the cluster node to which the interface belongs. Therefore, the interface identifier c/u, where c is the controller number and u is the unit number, now becomes n/c/u, where n is the node ID. For example, in the following figure, interface 1/2 of node n1 is represented as 0/1/2, interface 1/1 of node n2 is represented as 1/1/1, and interface 1/4 of node n3 is represented as 2/1/4.

Figure 1. Interface naming convention in a cluster



Server communication

The cluster communicates with the server through the physical connections between the cluster node and the server-side connecting device. The logical grouping of these physical connections is called the server data plane.

Client communication

The cluster communicates with the client through the physical connections between the cluster node and the client-side connecting device. The logical grouping of these physical connections is called the client data plane.

Inter-node communication

The cluster nodes communicate with each other by using the cluster backplane. The backplane is a set of connections in which one interface of each node is connected to a common switch, which is called the cluster backplane switch. Each node of the cluster uses a special MAC address to communicate with other nodes through the backplane.

The following figures shows the communication interfaces in L2 clusters and L3 clusters.

Figure 2. Cluster communication interfaces - L2 cluster

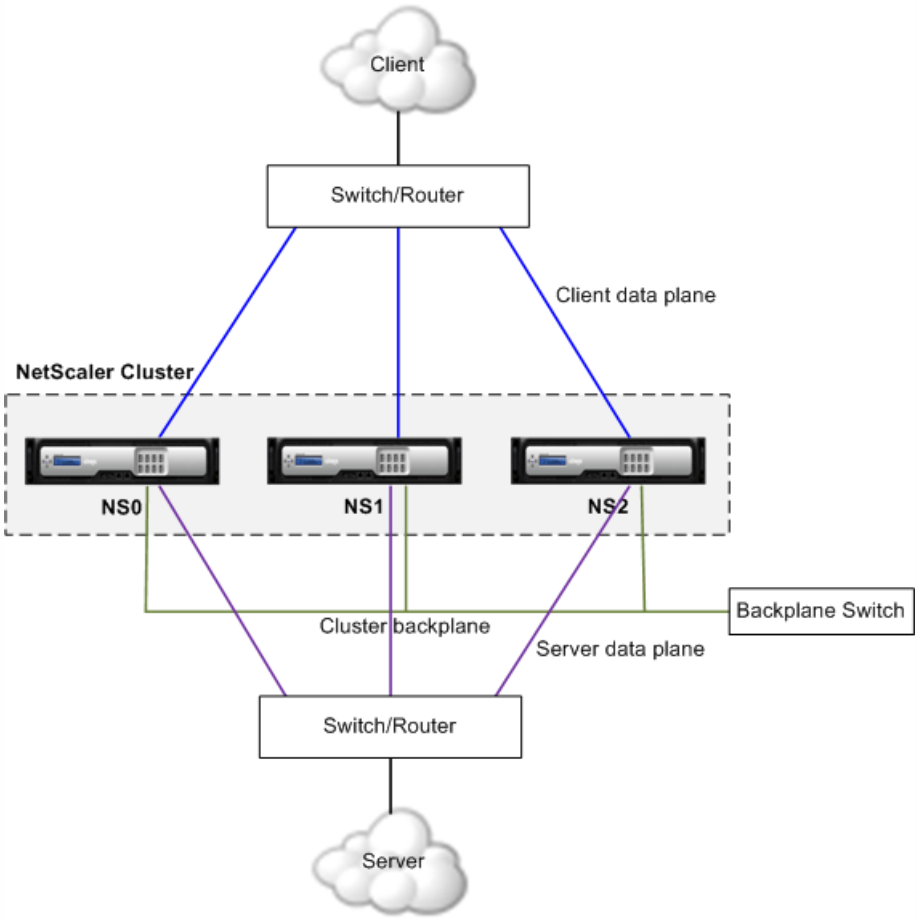
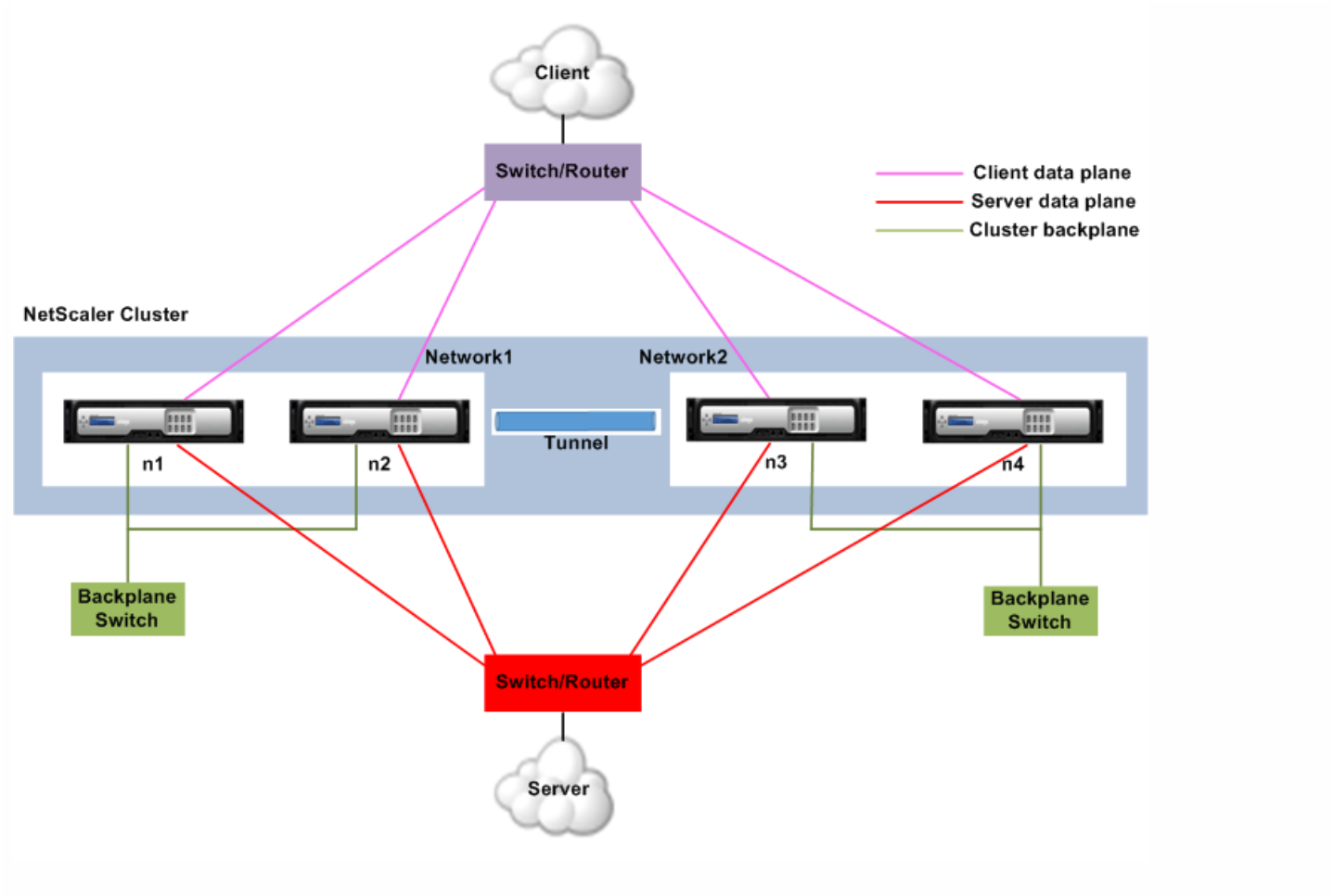


Figure 3. Cluster communication interfaces - L3 cluster



Traffic Distribution in a Cluster Setup

May 25, 2015

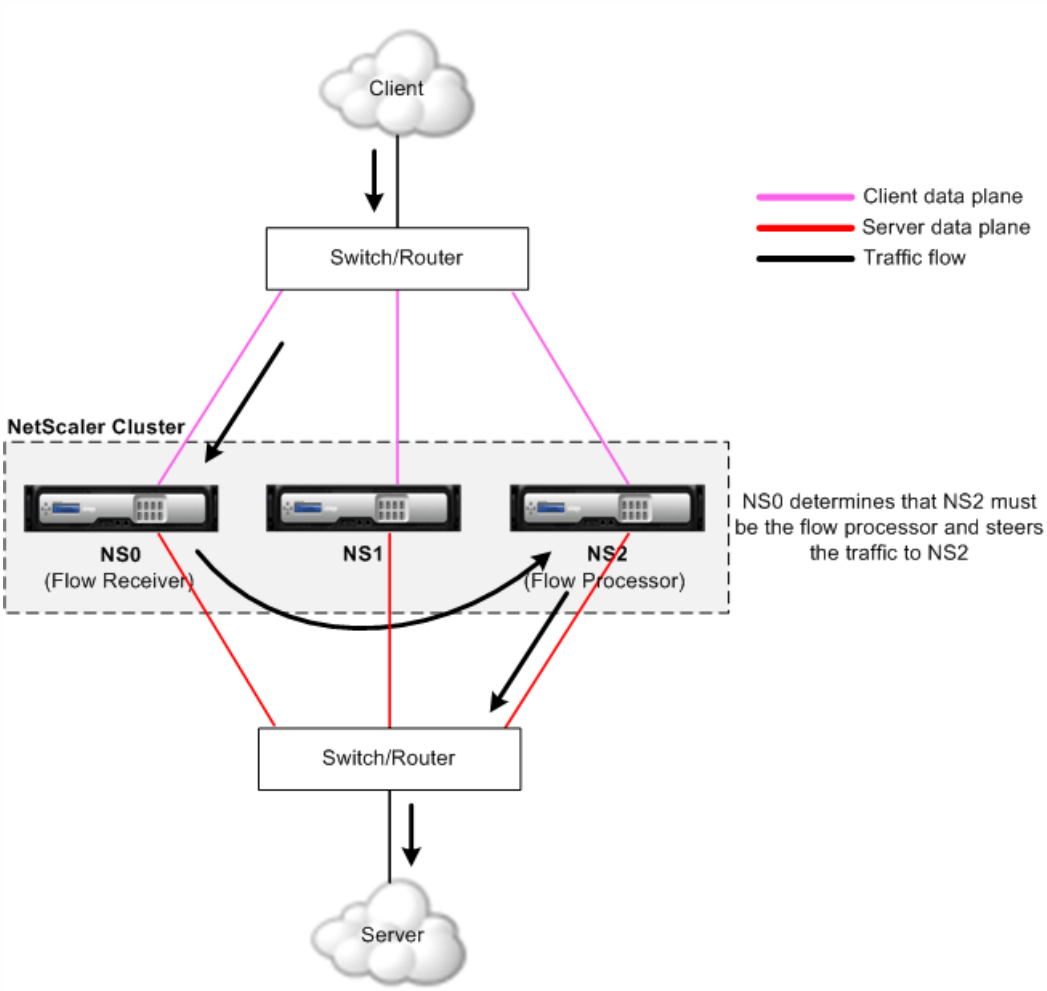
In a cluster setup, external networks view the collection of NetScaler appliances as a single entity. So, the cluster must select a single node that must receive the traffic. The cluster does this selection by using Equal Cost Multiple Path (ECMP) or cluster link aggregation traffic distribution mechanism. The selected node is called the flow receiver.

The flow receiver gets the traffic and then, using internal cluster logic determines the node that must process the traffic. This node is called the flow processor. The flow receiver steers the traffic to the flow processor over the backplane.

Note:

- The flow receiver and flow processor must be nodes capable of serving traffic.

Figure 1. Traffic distribution in a cluster



The above figure shows a client request flowing through the cluster. The client sends a request to a virtual IP (VIP) address. A traffic distribution mechanism configured on the client data plane selects one of the cluster nodes as the flow receiver. The flow receiver receives the traffic, determines the node that must process the traffic, and steers the request to that node (unless the flow receiver selects itself as the flow processor).

The flow processor establishes a connection with the server. The server processes the request and sends the response to the subnet IP (SNIP) address that sent the request to the server.

- If the SNIP address is a striped or partially striped IP address, the traffic distribution mechanism configured on the server data plane selects one of the cluster nodes as the flow receiver. The flow receiver receives the traffic, determines the flow processor, and steers the request to the flow processor through the cluster backplane.
- If the SNIP address is a spotted IP address, the node that owns the SNIP address receives the response from the server.

In an asymmetric cluster topology (all cluster nodes are not connected to the external switch), you must use linksets either exclusively or combined with ECMP or cluster link aggregation. For more information, see "Using Linksets".

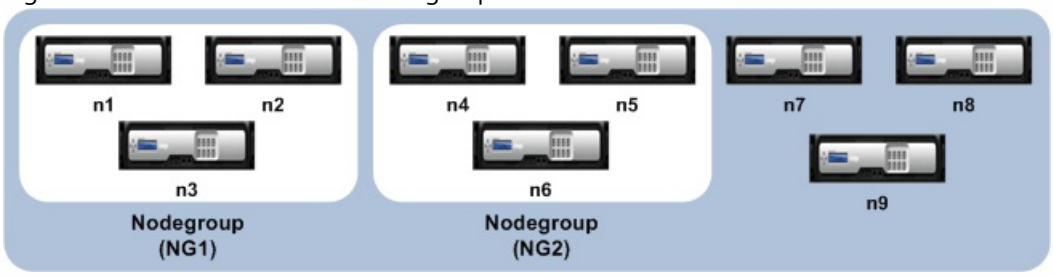
Cluster Nodegroups

May 25, 2015

Note: Nodegroups are supported from NetScaler 10.1 onwards.

As the name indicates, a cluster nodegroup is a group of cluster nodes.

Figure 1. NetScaler cluster with nodegroups



The above figure shows a cluster which has nodegroups NG1 and NG2 that include 3 cluster nodes each. The cluster also has 3 nodes that are not part of any nodegroup.

A nodegroup can be configured for the following:

- To define spotted and partially striped configurations. For more information, see "Nodegroups for Spotted and Partially-Striped Configurations".
- To configure redundancy of nodegroups. For more information, see "Configuring Redundancy for Nodegroups".
  Note: Supported from NetScaler 10.5 Build 52.1115.e onwards.

Note: The above functions of a nodegroup are mutually exclusive. This means that a nodegroup can provide only one of the above mentioned functionality.

Cluster and Node States

Jun 04, 2014

For a cluster to be functional, a majority of the nodes (n/2 + 1) must be online and be able to serve traffic by satisfying the following criteria:

- Admin state must be ACTIVE
- Operational state must be ACTIVE
- Health status must be UP

The following table describes the states of a cluster.

Type	Description
Admin	An admin state is configured when you add the node to the cluster. It indicates the purpose of the node, which can be in one of the following states: • **Active.** Nodes in this state serve traffic if they are operationally active and healthy. • **Passive.** Nodes in this state do not serve traffic but are in sync with the cluster. This is the default state of a cluster node. • **Spare.** Nodes in this state do not serve traffic but are in sync with the cluster. Spare nodes act as backup nodes for the cluster. If one of the nodes in the ACTIVE admin state becomes unavailable, a spare node becomes operationally active and starts serving traffic. Note: Whether the spare node remains operationally active depends on the preemption parameter of the add cluster instance command. If preemption is disabled, the spare node continues to serve traffic even if a node in ACTIVE admin state comes back online. If preemption is enabled, when a node in ACTIVE admin state comes back online, it preempts the spare node and starts serving traffic. The spare node goes back to inactive state.
Operational	When a node is part of a cluster, its operational state can change to ACTIVE, INACTIVE, or UNKNOWN. There are a number of reasons for a node being in INACTIVE or UNKNOWN state. Review the ns.log file or error counters to help determine the exact reason. Note: Passive nodes are always operationally INACTIVE. Spare nodes are ACTIVE only when they are serving traffic. Else, they are operationally INACTIVE.
Health	Depending on its health, a node can either be UP or NOT UP. To view the reasons for a node being in NOT UP state, run the show cluster node command for that node.

Routing in a Cluster

Apr 12, 2013

Routing in a cluster works in much the same way as routing in a standalone system. A few points to note:

- Routing runs only on spotted SNIP addresses and NSIP addresses.
- All routing configurations must be performed from the cluster IP address and the configurations are propagated to the other cluster nodes.
- Node-specific routing configurations must be performed by using the owner-node argument as follows:

```
!
interface vlan97
!
router ospf
 owner-node 0
  ospf router-id 97.131.0.1
 exit-owner-node
 owner-node 1
  ospf router-id 97.131.0.2
 exit-owner-node
 owner-node 2
  ospf router-id 97.131.0.3
 exit-owner-node
 redistribute kernel
 network 97.0.0.0/8 area 0
!
```

- Retrieve node-specific routing configurations by specifying the node(s) in the owner-node argument as follows:

```
> vtysh
ns# owner-node 0 1
ns(node-0 1)# show cluster state
ns(node-0 1)# exit-owner-node
```

- Clear node-specific routing configurations by specifying the node(s) in the owner-node argument as follows:

```
> vtysh
ns# owner-node 0 1
ns(node-0 1)# clear config
ns(node-0 1)# exit-owner-node
```

- Routing protocol daemons can run and adjacencies can be formed on active and inactive nodes of a cluster.
- Only active nodes advertise host routes to VIP addresses.
- Active and inactive nodes can learn dynamic routes and install them into the routing table.
- Routes learnt on a node are propagated to other nodes in the cluster only if route propagation is configured. This is mostly needed in asymmetric topologies where the unconnected nodes may not be able to form adjacencies.

  ```
  ns(config)# ns route-install propagate
  ```

  Note: Make sure that route propagation is not configured in a symmetric cluster topology as it can result in making the node unavailable to the cluster.

Setting up a NetScaler Cluster

Jun 15, 2015

NetScaler appliances that you want to add to the cluster must satisfy the criteria specified in "Prerequisites for Cluster Nodes". Before actually setting up a cluster, you must be aware of cluster basics. For information, see "Cluster Overview".

Forming a cluster requires you to set up inter-node communication, create the cluster (by adding the first NetScaler appliance), and then add the other cluster nodes. Each of these steps is explained with relevant details in subsequent topics.

Setting up Inter-Node Communication

Feb 04, 2016

The nodes in a cluster communicate with each other through the cluster backplane.

1. Identify the network interface that you want to use for the backplane.
2. Connect an Ethernet or optical cable from the selected network interface to the cluster backplane switch.

For example, to use interface 1/2 as the backplane interface for node 4, connect a cable from the 1/2 interface of node 4 to the backplane switch.

Important points to note when setting up the cluster backplane

- Do not use the appliance's management interface (0/1) as the backplane interface.
- Backlane interfaces must not be used for the client or server data planes.
- Configure a link aggregate (LA) channel to optimize the throughput of the cluster backplane.
- Citrix recommends that you dedicate a separate switch for the backplane, so that large amounts of traffic can be handled seamlessly.
- Backplane interfaces of all nodes of a cluster must be connected to the same switch and bound to the same L2 VLAN.
- If you have multiple clusters with the same cluster instance ID, make sure that the backplane interfaces of each cluster are bound to a different VLAN.
- The backplane interface is always monitored, regardless of the HA monitoring settings of that interface.
- The state of MAC spoofing on the different virtualization platforms can affect the steering mechanism on the cluster backplane. Therefore, make sure the appropriate state is configured:
  - XenServer - Disable MAC spoofing
  - HyperV - Enable MAC spoofing
  - ESX - Enable MAC spoofing (also make sure "Forged Transmits" is enabled)
- The Maximum Transmission Unit (MTU) for interfaces of the backplane switch must be greater than or equal to 1512 bytes, if features like MBF, L2 policies, ACLs, routing in CLAG deployments are configured. The MTU on the cluster backplane is automatically updated.

Creating a NetScaler Cluster

Apr 03, 2015

To create a cluster, start by taking one of the NetScaler appliances that you want to add to the cluster. On this node, you must create the cluster instance and define the cluster IP address. This node is the first cluster node and is called the cluster configuration coordinator. All configurations that are performed on the cluster IP address are stored on this node and then propagated to the other cluster nodes.

The responsibility of configuration coordination in a cluster is not fixed to a specific node. It can change over time depending on the following factors:

- The priority of the node. The node with the highest priority (lowest priority number) is made the configuration coordinator. Therefore, if a node with a priority number lower than that of the existing configuration coordinator is added, the new node takes over as the configuration coordinator.
  Note: Node priority can be configured from NetScaler 10.1 onwards.
- If the current configuration coordinator goes down. The node with the next lowest priority number takes over as the configuration coordinator. If the priority is not set or if there are multiple nodes with the lowest priority number, the configuration coordinator is selected from one of the available nodes.

Note: The configurations of the appliance (including SNIP addresses and VLANs) are cleared by implicitly executing the clear ns config extended command. However, the default VLAN and NSVLAN are not cleared from the appliance. Therefore, if you want the NSVLAN on the cluster, make sure it is created before the appliance is added to the cluster.

1. Log on to an appliance (for example, appliance with NSIP address 10.102.29.60) that you want to add to the cluster.
2. Add a cluster instance.
   add cluster instance <clId>

   Note:
   - The cluster instance ID must be unique within a LAN.
3. Add the NetScaler appliance to the cluster.
   add cluster node <nodeId> <IPAddress> -state <state> -backplane <interface_name>

   Example

   Adding a node for an L2 cluster (all cluster nodes are in the same network).

   ```
   > add cluster node 0 10.102.29.60 -state PASSIVE -backplane 0/1/1
   ```
4. Add the cluster IP address (for example, 10.102.29.61) on this node.
   add ns ip <IPAddress> <netmask> -type clip

   Example
   ```
   > add ns ip 10.102.29.61 255.255.255.255 -type clip
   ```
5. Enable the cluster instance.
   enable cluster instance <clId>

6. Save the configuration.
   save ns config

7. Warm reboot the appliance.

reboot -warm

Verify the cluster configurations by using the show cluster instance command. Verify that the output of the command displays the NSIP address of the appliance as a node of the cluster.

1. Log on to an appliance (for example, an appliance with NSIP address 10.102.29.60) that you intend to add to the cluster.
2. Navigate to System > Cluster.
3. In the details pane, click the Manage Cluster link.
4. In the Cluster Configuration dialog box, set the parameters required to create a cluster. For a description of a parameter, hover the mouse cursor over the corresponding text box.
5. Click Create.
6. In the Configure cluster instance dialog box, make sure that the Enable cluster instance check box is selected.
7. In the Cluster Nodes pane, select the node and click Open.
8. In the Configure Cluster Node dialog box, set the State.
9. Click OK, and then click Save.
10. Warm reboot the appliance.

Adding a Node to the Cluster

Apr 03, 2015

You can seamlessly scale the size of a cluster to include a maximum of 32 nodes. When a NetScaler appliance is added to the cluster, the configurations from that appliance are cleared and cluster configurations are synchronized on this node. There can be an intermittent drop in traffic while the synchronization is in progress.

Note:

- The licenses of the appliance are checked against the licenses available on the configuration coordinator. The appliance is added if the licenses match.
- If you use the NetScaler CLI to add a node, the new node does not become a functional part of the cluster until it is explicitly joined to the cluster. Therefore, after adding the node, log on to that node and join the node to the cluster. Alternatively, you can add the node from the command line and use the configuration utility to join the node to the cluster. If you use the configuration utility, you need only log on to the cluster IP address and add the node. The newly added node is automatically joined to the cluster.

Important: Before you add a NetScaler appliance to a cluster:

- Set up the backplane interface for the node.
- If you want the NSVLAN on the cluster, make sure that the NSVLAN is created on the appliance before it is added to the cluster.
- Citrix recommends that you add the node as a passive node. Then, after joining the node to the cluster, complete the node specific configuration from the cluster IP address. Run the force cluster sync command if the cluster has only spotted IP addresses, has L3 VLAN binding, or has static routes.
- When an appliance with a preconfigured link aggregate (LA) channel is added to a cluster, the LA channel continues to exist in the cluster environment. The LA channel is renamed from LA/x to nodeId/LA/x, where LA/x is the LA channel identifier.

To add a node to the cluster by using the command line interface

1. Log on to the cluster IP address and, at the command prompt, do the following:
   1. Add the appliance (for example, 10.102.29.70) to the cluster.
      add cluster node <nodeId> <IPAddress> -state <state> -backplane <interface_name>

      Example

      ```
      > add cluster node 1 10.102.29.70 -state PASSIVE -backplane 1/1/1
      ```
   2. Save the configuration.
      save ns config

2. Log on to the newly added node (for example, 10.102.29.70) and do the following:
   1. Join the node to the cluster.
      join cluster -clip <ip_addr> -password <password>

      Example

      ```
      > join cluster -clip 10.102.29.61 -password nsroot
      ```
   2. Save the configuration.
      save ns config

   3. Warm reboot the appliance.

```
reboot -warm
```

To add a node to the cluster by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster > Nodes.
3. In the details pane, click Add to add the new node (for example, 10.102.29.70).
4. In the Create Cluster Node dialog box, configure the new node. For a description of a parameter, hover the mouse cursor over the corresponding text box.
5. Click Create. When prompted to perform a warm reboot, click Yes.

To join a previously added node to the cluster by using the configuration utility

If you have used the command line to add a node to the cluster, but have not joined the node to the cluster, you can use the following procedure.

Note: When a node joins the cluster, it takes over its share of traffic from the cluster and hence an existing connection can get terminated.

1. Log on to the node that you want to join to the cluster (for example, 10.102.29.70).
2. Navigate to System > Cluster.
3. In the details pane, under Get Started, click the Join Cluster link.
4. In the Join to existing cluster dialog box, set the cluster IP address and the nsroot password of the configuration coordinator. For a description of a parameter, hover the mouse cursor over the corresponding text box.
5. Click OK.

Viewing the Details of a Cluster

Feb 13, 2015

You can view the details of the cluster instance and the cluster nodes by logging on to the cluster IP address.

To view details of a cluster instance by using the command line interface

Log on to the cluster IP address and, at the command prompt, type:

show cluster instance <clId>

Note: When executed from the NSIP address of a cluster node that is not the configuration coordinator, this command displays the status of the cluster on this node.

To view details of a cluster node by using the command line interface

Log on to the cluster IP address and, at the command prompt, type:

show cluster node <nodeId>

To view details of a cluster instance by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster.
3. In the details pane, under Get Started, click the Manage Cluster link to view the details of the cluster.

To view details of a cluster node by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster > Nodes.
3. In the details pane, click the node for which you want to view the details.

Distributing Traffic Across Cluster Nodes

Feb 13, 2015

After you have created the NetScaler cluster and performed the required configurations, you must deploy Equal Cost Multiple Path (ECMP) or cluster Link Aggregation (LA) on the client data plane (for client traffic) or server data plane (for server traffic). These mechanisms distribute external traffic across the cluster nodes.

Using Equal Cost Multiple Path (ECMP)

Feb 13, 2015

With the Equal Cost Multiple Path (ECMP) mechanism, the router has equal-cost routes to VIP addresses with the next hops as the active nodes of the cluster. The router uses a stateless hash-based mechanism to distribute traffic across the routes.

Note: Routes are limited to the maximum number of ECMP routes supported by the upstream router.
To use ECMP, you must first enable the required routing protocol (OSPF, RIP, BGP, or ISIS) on the cluster IP address. You must bind the interfaces and the spotted IP address (with dynamic routing enabled) to a VLAN. Configure the selected routing protocol and redistribute the kernel routes on the ZebOS by using the vtysh shell.

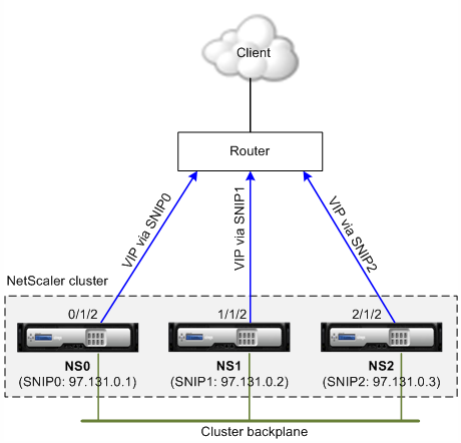
You must perform similar configurations on the cluster IP address and on the external connecting device.

Note:

- All routing configurations must be done through the cluster IP address. No configurations must be performed on individual cluster nodes.
- Make sure that the licenses on the cluster support dynamic routing, otherwise ECMP does not work.
- ECMP is not supported for wildcard virtual servers since RHI needs a VIP address to advertise to a router and wildcard virtual servers do not have associated VIP addresses.

You must have detailed knowledge of routing protocols to use ECMP. For more information, see "Configuring Dynamic Routes. For more information on routing in a cluster, see "Routing in a Cluster".

Figure 1. ECMP topology



As seen in the above figure, the ECMP router can reach the VIP address via SNIP0, SNIP1, or SNIP2.

To configure ECMP on the cluster by using the command line interface

1. Log on to the cluster IP address.
2. Enable the routing protocol.
   enable ns feature <feature>

   Example: To enable the OSPF routing protocol.
   `> enable ns feature ospf`
3. Add a VLAN.
   add vlan <id>

   Example
   `> add vlan 97`
4. Bind the interfaces of the cluster nodes to the VLAN.
   bind vlan <id> -ifnum <interface_name>

   Example
   `> bind vlan 97 -ifnum 0/1/2 1/1/2 2/1/2`
5. Add a spotted SNIP address for each node and enable dynamic routing on it.
   add ns ip <SNIP> <netmask> -ownerNode <positive_integer> -dynamicRouting ENABLED

   Example
   `> add ns ip 97.131.0.1 255.0.0.0 -ownerNode 0 -dynamicRouting ENABLED -type SNIP > add ns ip 97.131.0.2 255.0.0.0 -ownerNode 1 -dynamicRouting ENABLED -type SI`
6. Bind one of the spotted SNIP addresses to the VLAN. When you bind one spotted SNIP address to a VLAN, all other spotted SNIP addresses defined on the cluster in that subnet are automatically bound to the VLAN.
   bind vlan <id> -IPAddress <SNIP> <netmask>

   Example
   `> bind vlan 97 -ipAddress 97.131.0.1 255.0.0.0`
   Note: You can use NSIP addresses of the cluster nodes instead of adding SNIP addresses. If so, you do not have to perform steps 3 - 6.
7. Configure the routing protocol on ZebOS using vtysh shell.
   Example: To configure OSPF routing protocol on node IDs 0, 1, and 2.

   `> vtysh ! interface vlan97 ! router ospf owner-node 0 ospf router-id 97.131.0.1 exit-owner-node owner-node 1 ospf router-id 97.131.0.2 exit-owner-node owne`
   Note: For VIP addresses to be advertised, RHI setting must done by using the vserverRHILevel parameter as follows:
   add ns ip <IPAddress> <netmask> -type VIP -vserverRHILevel <vserverRHILevel>

For OSPF specific RHI settings, there are additional settings that can be done as follows:

add ns ip <IPAddress> <netmask> -type VIP -ospfLSAType (TYPE1 | TYPE5) -ospfArea <positive_integer>

Use the add ns ip6 command to perform the above commands on IPv6 addresses.

8. Configure ECMP on the external switch. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1) switch. Similar configurations must be performed on other switches.
   //For OSPF (IPv4 addresses) Global config: Configure terminal feature ospf Interface config: Configure terminal interface Vlan10 no shutdown ip address 97.131

Use Case: ECMP with BGP Routing

Aug 06, 2013

To configure ECMP with BGP routing protocol, perform the following steps:

1. Log on to the cluster IP address.
2. Enable BGP routing protocol.
   ```
   > enable ns feature bgp
   ```
3. Add VLAN and bind the required interfaces.
   ```
   > add vlan 985
   > bind vlan 985 -ifnum 0/0/1 1/0/1
   ```
4. Add the spotted IP address and bind them to the VLAN.
   ```
   > add ns ip 10.100.26.14 255.255.255.0 -ownerNode 1 -dynamicRouting ENABLED
   > add ns ip 10.100.26.15 255.255.255.0 -ownerNode 2 -dynamicRouting ENABLED
   > bind vlan 985 -ipAddress 10.100.26.10 255.255.255.0
   ```
5. Configure BGP routing protocol on ZebOS using vtysh shell.
   ```
   > vtysh
   conf t
   router bgp 65535
   neighbor 10.100.26.1 remote-as  65535
   ```
6. Configure BGP on the external switch. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1) switch. Similar configurations must be performed on other switches.
   ```
   router bgp 65535
   no synchronization
   bgp log-neighbor-changes
   neighbor 10.100.26.14 remote-as 65535
   neighbor 10.100.26.15 remote-as 65535
     no auto-summary
   dont-capability-negotiate
   dont-capability-negotiate
   no dynamic-capability
   ```

Using Cluster Link Aggregation

Feb 08, 2016

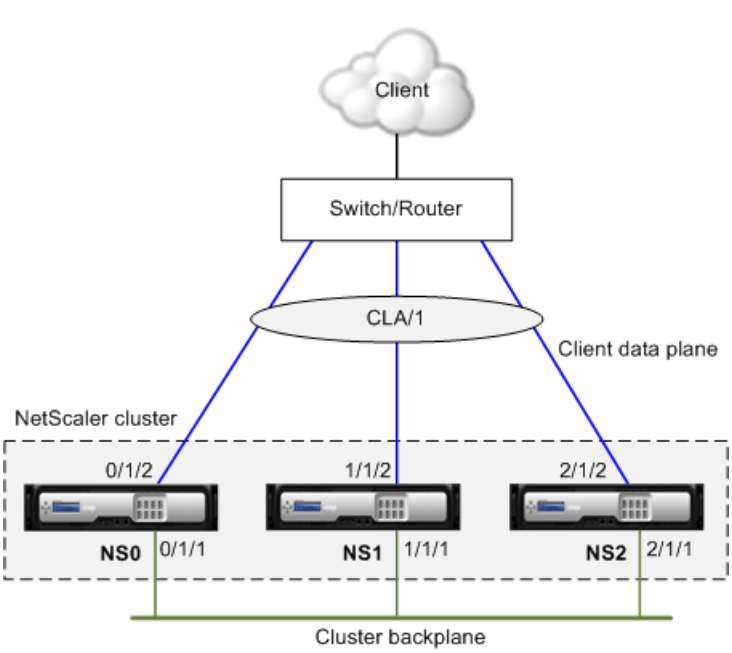
Cluster link aggregation, as the name suggests, is a group of interfaces of cluster nodes. It is an extension of NetScaler link aggregation. The only difference is that, while link aggregation requires the interfaces to be from the same device, in cluster link aggregation, the interfaces are from different nodes of the cluster.

Note: Cluster link aggregation is supported only for a cluster of hardware (MPX) NetScaler appliances.
For more information about link aggregation, see "Configuring Link Aggregation".

Cluster link aggregation can be either static or dynamic.

For example, consider a three-node cluster where all three nodes are connected to the upstream switch. A cluster LA channel (CLA/1) is formed by binding interfaces 0/1/2, 1/1/2, and 2/1/2.

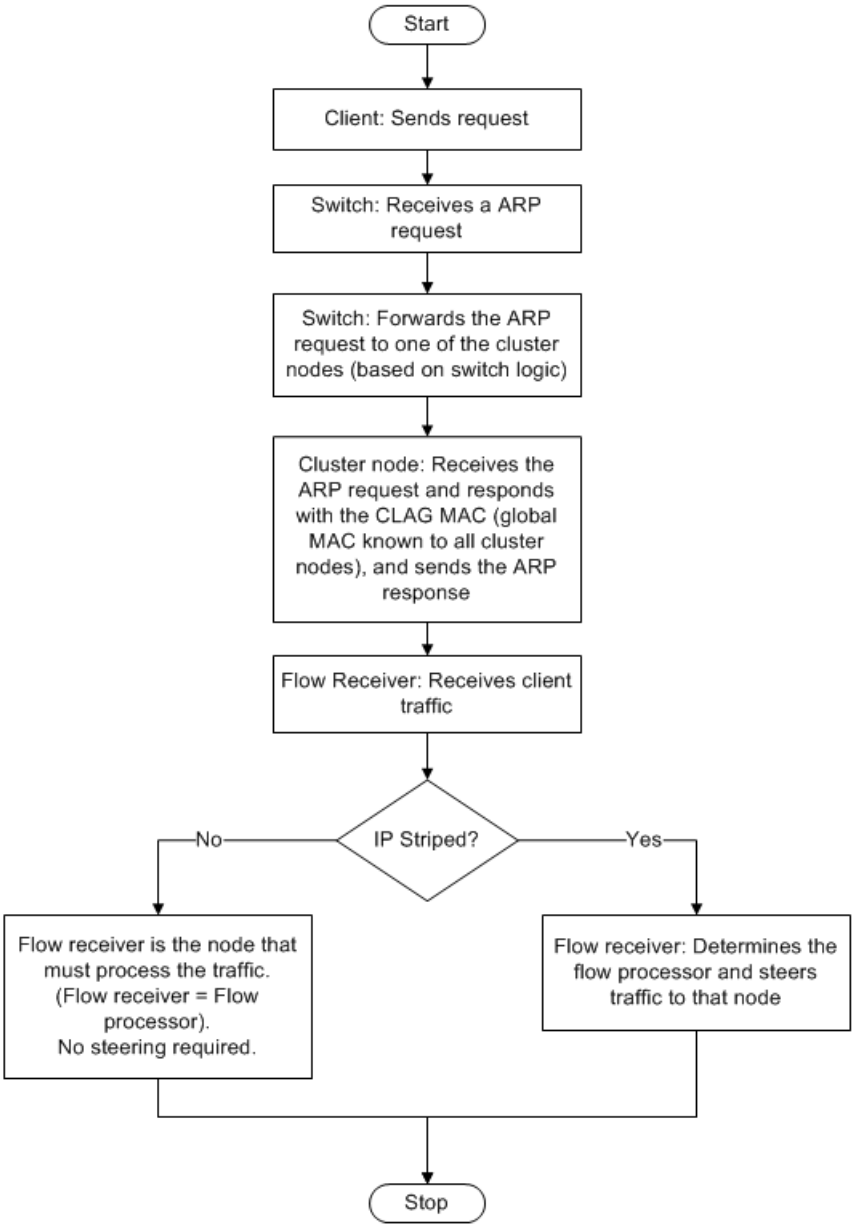
Figure 1. Cluster Link Aggregation topology



A cluster LA channel has the following attributes:

- Each channel has a unique MAC agreed upon by cluster nodes.
- The channel can bind both local and remote nodes' interfaces.
- A maximum of four cluster LA channels are supported in a cluster.
- Backplane interfaces cannot be part of a cluster LA channel.
- When an interface is bound to a cluster LA channel, the channel parameters have precedence over the network interface parameters. A network interface can be bound to one channel only.
- Management access to a cluster node, must not be configured on a cluster LA channel (for example, CLA/1) or its member interfaces. This is because when the node is INACTIVE, the corresponding cluster LA interface is marked as power down and therefore looses management access.

Figure 2. Traffic distribution flow using cluster LA



Static Cluster Link Aggregation

Feb 13, 2015

You must configure a static cluster LA channel on the cluster IP address and on the external connecting device. If possible, configure the upstream switch to distribute traffic based on IP address or port instead of MAC address.

For more information about configuring a static LA channel, see "Configuring Link Aggregation Manually".

To configure a static cluster LA channel by using the command line interface

1. Log on to the cluster IP address.
   Note: Make sure that you configure the cluster LA channel on the cluster IP address before configuring link aggregation on the external switch. Otherwise, the switch will forward traffic to the cluster even though the cluster LA channel is not configured. This can lead to loss of traffic.
2. Create a cluster LA channel.
   add channel <id> -speed <speed>

   Example
   > add channel CLA/1 -speed 1000
   Note: You must not specify the speed as AUTO. Rather, you must explicitly specify the speed as 10, 100, 1000, or 10000. Only interfaces that have the speed matching the <speed> attribute in the cluster LA channel are added to the active distribution list.
3. Bind the required interfaces to the cluster LA channel. Make sure that the interfaces are not used for the cluster backplane.
   bind channel <id> <ifnum>

   Example
   > bind channel CLA/1 0/1/2 1/1/2 2/1/2
4. Verify the configurations.
   show channel <id>

   Example
   > show channel CLA/1
   Note: You can bind the cluster LA channel to a VLAN by using the bind vlan command. The interfaces of the channel are automatically bound to the VLAN.
5. Configure static LA on the external switch. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1). Similar configurations must be performed on other switches.
   Global config:
   Configure terminal

   Interface level config:

   ```
   interface Ethernet2/47
     switchport
     switchport access vlan 10
     channel-group 7 mode on
     no shutdown
   ```

```
interface Ethernet2/48
  switchport
  switchport access vlan 10
  channel-group 7 mode on
  no shutdown
```

Dynamic Cluster Link Aggregation

Apr 29, 2015

Dynamic cluster LA channel uses Link Aggregation Control Protocol (LACP). For more information about configuring a dynamic LA channel, see "Configuring Link Aggregation by Using the Link Aggregation Control Protocol".

You must perform similar configurations on the cluster IP address and on the external connecting device. If possible, configure the upstream switch to distribute traffic based on IP address or port instead of MAC address.

Points to remember:

- Enable LACP (by specifying the LACP mode as either ACTIVE or PASSIVE).

  Note: Make sure the LACP mode is not set as PASSIVE on both the NetScaler cluster and the external connecting device.
- Specify the same LACP key on each interface that you want to be the part of the channel. For creating a cluster LA channel, the LACP key can have a value from 5 through 8. For example, if you set the LACP key on interfaces 0/1/2, 1/1/2, and 2/1/2 to 5, CLA/1 is created. The interfaces 0/1/2, 1/1/2, and 2/1/2 are automatically bound to CLA/1. Similarly, if you set the LACP key to 6, CLA/2 channel is created.
- Specify the LAG type as Cluster.

To configure a dynamic cluster LA channel by using the command line interface

On the cluster IP address, for each interface that you want to add to the cluster LA channel, type:

set interface <id> -lacpMode <lacpMode> -lacpKey <positive_integer> -lagType CLUSTER

Example: To configure a cluster LA channel CLA/1 of 3 interfaces.

```
> set interface 0/1/2 -lacpMode active -lacpKey 5 -lagType Cluster
> set interface 1/1/2 -lacpMode active -lacpKey 5 -lagType Cluster
> set interface 2/1/2 -lacpMode active -lacpKey 5 -lagType Cluster
```

Note: Optionally, you can enable Link Redundancy in a Cluster with LACP.

Similarly, configure dynamic LA on the external switch. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1). Similar configurations must be performed on other switches.

```
Global config:
Configure terminal
feature lacp
Interface level config:

interface Ethernet2/47
   switchport
   switchport access vlan 10
   channel-group 7 mode active
   no shutdown

interface Ethernet2/48
   switchport
```

```
switchport access vlan 10
channel-group 7 mode active
no shutdown
```

Using Linksets

Feb 27, 2015

Linksets must be used when some cluster nodes are not physically connected to the external network. In such a cluster topology, the unconnected cluster nodes use the interfaces specified in the linkset to communicate with the external network through the cluster backplane. Linksets are typically used in scenarios when the connecting devices have insufficient ports to connect the cluster nodes.

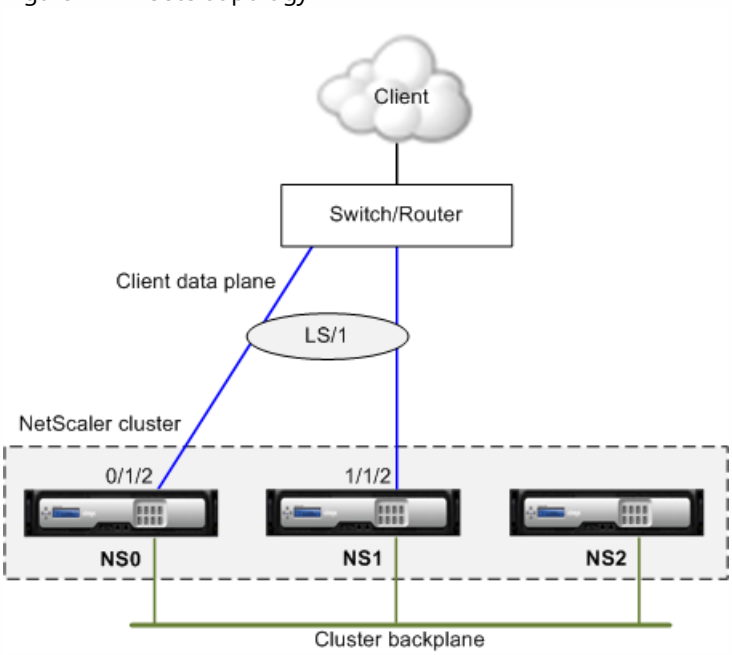
Note: Linksets are a mandatory configuration in the following scenarios:

- For deployments that require MAC-Based Forwarding (MBF).
- To improve manageability of ACL and L2 policies involving interfaces. You must define a linkset of the interfaces and add ACL and L2 policies based on linksets.

Linksets must be configured only through the cluster IP address.

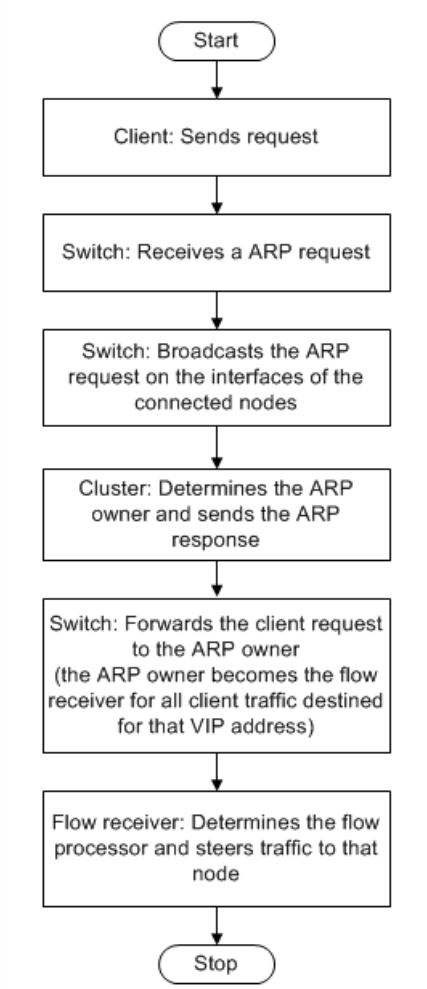
For example, consider a three node cluster where the upstream switch has only two ports available. Using linksets, you can connect two nodes to the switch and leave the third node unconnected. In the following figure, a linkset (LS/1) is formed by binding the interfaces 0/1/2 and 1/1/2. NS2 is the unconnected node of the cluster.

Figure 1. Linksets topology



The linkset informs NS2 that it can use interfaces 0/1/2 and 1/1/2 to communicate with the network devices. All traffic to and from NS2 is now routed through interfaces 0/1/2 or 1/1/2.

Figure 2. Traffic distribution flow using linksets



To configure a linkset by using the command line interface

1. Log on to the cluster IP address.
2. Create a linkset.
   add linkset <id>

   Example
   > add linkset LS/1
3. Bind the required interfaces to the linkset. Make sure the interfaces are not used for the cluster backplane.
   bind linkset <id> -ifnum <interface_name> ...

   Example
   > bind linkset LS/1 -ifnum 0/1/2 1/1/2
4. Verify the linkset configurations.
   show linkset <id>

   Example
   > show linkset LS/1
   Note: You can bind the linkset to a VLAN by using the bind vlan command. The interfaces of the linkset are automatically bound to the VLAN.

To configure a linkset by using the configuration utility

1. Log on to the cluster IP address.

2. Navigate to System > Network > Linksets.
3. In the details pane, click Add.
4. In the Create Linkset dialog box:
    1. Specify the name of the linkset by setting the Linkset parameter.
    2. Specify the Interfaces to be added to the linkset and click Add. Repeat this step for each interface you want to add to the linkset.
5. Click Create, and then click Close.

Managing the NetScaler Cluster

Nov 17, 2014

After you have created a cluster and configured the required traffic distribution mechanism, the cluster is able to serve traffic. During the lifetime of the cluster, you can perform cluster tasks such as configuring nodegroups, disabling nodes of a cluster, discovering NetScaler appliances, viewing statistics, synchronizing cluster configurations, cluster files, and the time across the nodes, and upgrading or downgrading the software of cluster nodes.

Nodegroups for Spotted and Partially-Striped Configurations

May 25, 2015

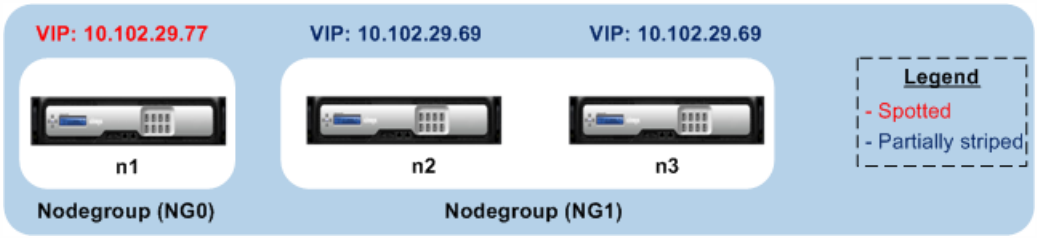
By virtue of the default cluster behavior, all configurations performed on the cluster IP address are available on all nodes of the cluster. However, there might be cases where you need some configurations to be available only on specific cluster nodes.

You can achieve this requirement by defining a nodegroup that includes the specific cluster nodes, and then binding the configuration to that nodegroup. This ensures that the configuration is active only on those cluster nodes. These configurations are called partially-striped or spotted (if active only one a single node). For more information, see Striped, Partially Striped, and Spotted Configurations.

For example, consider a cluster with three nodes. You create a nodegroup NG0 that includes node n1 and another nodegroup NG1 that includes n2 and n3. Bind load balancing virtual servers .77 to NG0 and load balancing virtual server .69 to NG1.

This means that virtual server .77 will be active only on n1 and consequently only n1 will receive traffic that is directed to .77. Similarly, virtual server .69 will be active only on nodes n2 and n3 and consequently only n2 and n3 will receive traffic that is directed to .69.

Figure 1. NetScaler cluster with nodegroups configured for spotted and partial-striped configurations



The entities or configurations that you can bind to a nodegroup are:

- Load balancing, content switching, cache redirection, authentication (AAA) virtual servers
  Note: FTP load balancing virtual servers cannot be bound to nodegroups.
- VPN virtual server (Supported from NetScaler 10.5 Build 50.10 onwards)
- Global Server Load Balancing (GSLB) sites and other GSLB entities (Supported from NetScaler 10.5 Build 52.11 onwards)
- Limit identifiers and stream identifiers

Behavior of Nodegroups

Mar 18, 2015

Due to the interoperability of nodegroups with different NetScaler features and entities, there are some behavioral aspects to be noted. Nodes in a nodegroup can also be backed up. Read on for more information.

General behavior of a cluster nodegroup

- A nodegroup that has entities bound to it cannot be removed.
- A cluster node that belongs to a nodegroup with entities bound to it, cannot be removed.
- A cluster instance that has nodegroups with entities bound to it, cannot be removed.
- You cannot add an entity that has a dependency on another entity that is not part of the nodegroup. If you need to do so, first remove the dependency. Then, add both the entities to the nodegroup and reassociate the entities.
  Examples:
  - Assume you have a virtual server, VS1, whose backup is virtual server VS2. To add VS1 to a nodegroup, first make sure that VS2 is removed as the backup server of VS1. Then, bind each server individually to the nodegroup, and then configure VS2 as the backup for VS1.
  - Assume you have a content switching virtual server, CSVS1, whose target load balancing virtual server is LBVS1. To add CSVS1 to a nodegroup, first remove LBVS1 as the target. Then, bind each server individually to the nodegroup, and then configure LBVS1 as the target.
  - Assume you have a load balancing virtual server, LBVS1, that has a policy which invokes another load balancing virtual server, LBVS2. To add either one of the virtual servers, first remove the association. Then, bind each server individually to the nodegroup, and then reassociate the virtual servers.
- You cannot bind an entity to a nodegroup that has no nodes and that has the `strict` option enabled. Consequently, you cannot unbind the last node of a nodegroup that has entities bound to it and that has the `strict` option enabled
- The `strict` option cannot be modified for a nodegroup that has no nodes but has entities bound to it.

Backing up Nodes in a Nodegroup

By default, a nodegroup is designed to provide back up nodes for members of a nodegroup. If a nodegroup member goes down, a cluster node that is not a member of the nodegroup dynamically replaces the failed node. This node is called the replacement node.

Note: For a single-member nodegroup, a backup node is automatically preselected when an entity is bound to the nodegroup.
When the original member of the nodegroup comes up, the replacement node, by default, is replaced by the original member node.

From NetScaler 10.5 Build 50.10 onwards, however, the NetScaler allows you to change this replacement behavior. When you enable the sticky option, the replacement node is retained even after the original member node comes up. The original node takes over only when the replacement node goes down.

You can also disable the backup functionality. To do this, you must enable the strict option. In this scenario, when a nodegroup member goes down, no other cluster node is picked up as a backup node. The original node continues being part of the nodegroup when it comes up. This option ensures that entities bound to a nodegroup are active only on nodegroup members.

Note: The strict and sticky option can be set only when creating a nodegroup.

Configuring Nodegroups for Spotted and Partially-Striped Configurations

Mar 31, 2015

To configure a nodegroup for spotted and partially-striped configurations you must first create a nodegroup and then bind the required nodes to the nodegroup. You must then associate the required entities to that nodegroup. The entities that are bound to the nodegroup will be:

- Spotted - If bound to a nodegroup that has a single node.
- Partially striped - If bound to a nodegroup that has more than one node.

Some points to remember:

- GSLB is supported on a cluster only when GSLB sites are bound to nodegroups that have a single cluster node. For more information, see Setting Up GSLB in a Cluster.
- NetScaler Gateway is supported on a cluster only when the VPN virtual servers are bound to nodegroups that have a single cluster node. The sticky option must be enabled on the nodegroup.
- Application firewall is supported on a cluster only when application firewall profiles are associated with virtual servers that are bound to nodegroups that have a single cluster node. You are not allowed to do the following:
  - Bind application firewall profiles to striped or partially striped virtual servers.
  - Bind the policy to a global bind point or to user-defined policy labels.
  - Unbind, from a nodegroup, a virtual server that has application firewall profiles.

Check NetScaler Features Supported in a Cluster to see the NetScaler versions from which GSLB, NetScaler Gateway, and application firewall are supported in a cluster.

To configure a nodegroup by using the command line interface

1. Log on to the cluster IP address.
2. Create a nodegroup. Type:
   add cluster nodegroup <name> -strict (**YES** | **NO**)

   Example
   ```
   > add cluster nodegroup NG0 -strict YES
   ```
3. Bind the required nodes to the nodegroup. Type the following command for each member of the nodegroup:
   bind cluster nodegroup <name> -node <nodeId>

   Example: To bind nodes with IDs 1, 5, and 6.
   ```
   > bind cluster nodegroup NG0 -node 1
   > bind cluster nodegroup NG0 -node 5
   > bind cluster nodegroup NG0 -node 6
   ```
4. Bind the entity to the nodegroup. Type the following command once for every entity that you want to bind:
   bind cluster nodegroup <name> (-vServer <string> | -identifierName <string> | -gslbSite <string> -service <string>)

   Note: The gslbSite and service parameters are available from NetScaler 10.5 onwards.
   Example: To bind virtual servers VS1 and VS2 and rate limit identifier named identifier1.
   ```
   > bind cluster nodegroup NG0 -vServer VS1
   > bind cluster nodegroup NG0 -vServer VS2
   > bind cluster nodegroup NG0 -identifierName identifier1
   ```

5. Verify the configurations by viewing the details of the nodegroup. Type:
   show cluster nodegroup <name>

   Example
   ```
   > show cluster nodegroup NG0
   ```

To configure a nodegroup by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster > Node Groups.
3. In the details pane, click Add.
4. In the Create Node Group dialog box, configure the nodegroup:
   1. Under Cluster Nodes, click the Add button.
      - The Available list displays the nodes that you can bind to the nodegroup and the Configured list displays the nodes that are bound to the nodegroup.
      - Click the + sign in the Available list to bind the node. Similarly, click the - sign in the Configured list to unbind the node.
   2. Under Virtual Servers, select the tab corresponding to the type of virtual server that you want to bind to the nodegroup. Click the Add button.
      - The Available list displays the virtual servers that you can bind to the nodegroup and the Configured list displays the virtual servers that are bound to the nodegroup.
      - Click the + sign in the Available list to bind the virtual server. Similarly, click the - sign in the Configured list to unbind the virtual server.

Configuring Redundancy for Nodegroups

Mar 18, 2015

Note: Supported from NetScaler 10.5 Build 52.1115.e onwards.

Nodegroups can be configured such that when one nodegroup goes down, another nodegroup can take over and process traffic. For example, when a nodegroup NG1 goes down, NG2 takes over.

Note: This functionality can be used to configure datacenter redundancy where each nodegroup is configured as a datacenter.

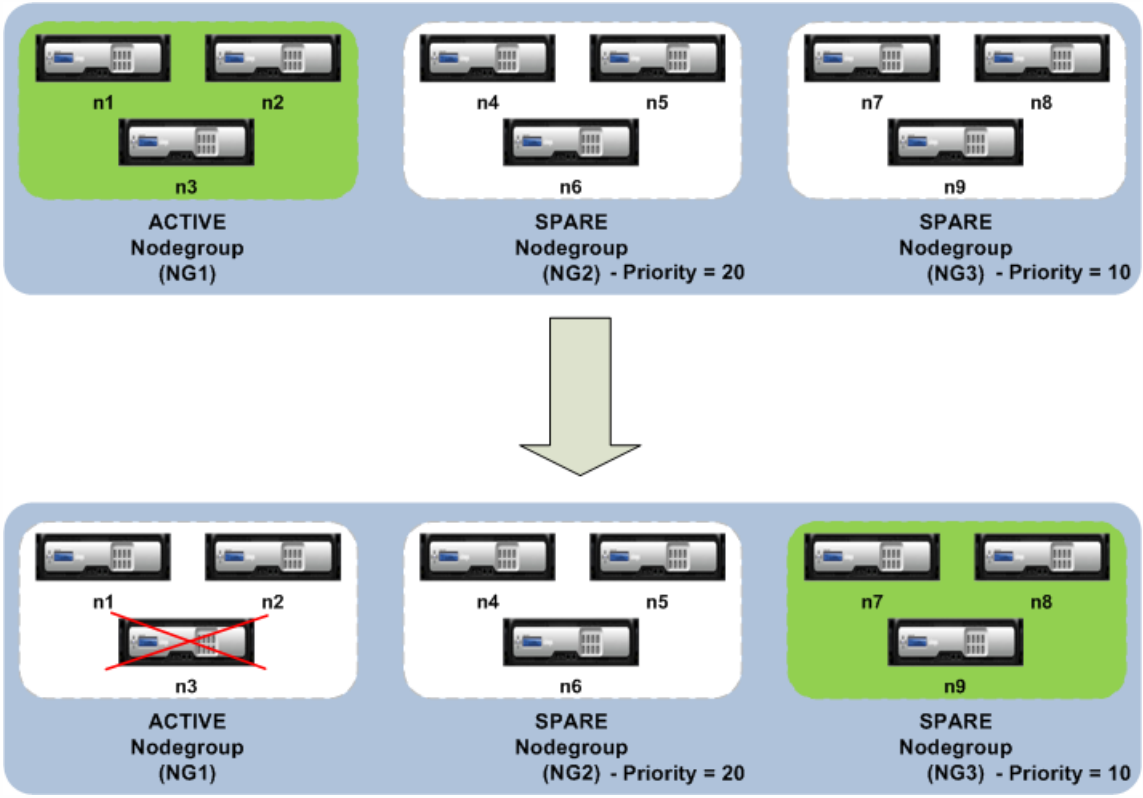
To achieve this use case, cluster nodes must be logically grouped into nodegroups, where some nodegroups must be configured as ACTIVE and others as SPARE. The active nodegroup with the highest priority (that is, the lowest priority number) is made operationally active and therefore serves traffic. When a node from this operationally active nodegroup goes down, the node count of this nodegroup is compared with the node count of the other active nodegroups in order of their priority. If a nodegroup has a higher or equal node count, that nodegroup is made operationally active. Else, the spare nodegroups are checked.

Note:

- Only one state-specific nodegroup can be active at a given point in time.
- A cluster node inherits the state of the nodegroup. So, if a node with "SPARE" state is added to nodegroup with state as "ACTIVE", the node automatically behaves as an active node.
- The preemption parameter that is defined for the cluster instance decides whether the initial active nodegroup will take control when the it comes up again.

The following figure shows a nodegroup setup that has nodegroup redundancy defined. NG1 is initially the active nodegroup. When it loses one of the nodes, the spare nodegroup (NG3) with the highest priority starts serving traffic.

Figure 1. NetScaler cluster with nodegroup redundancy configured



Configuring redundancy for nodegroups

1. Log on to the cluster IP address.
2. Create the active nodegroup and bind the required cluster nodes.
   add cluster nodegroup NG1 -state ACTIVE

   bind cluster nodegroup NG1 -node n1

   bind cluster nodegroup NG1 -node n2

   bind cluster nodegroup NG1 -node n3

3. Create the spare nodegroup and bind the requisite nodes.
   add cluster nodegroup NG2 -state SPARE -priority 20

   bind cluster nodegroup NG2 -node n4

   bind cluster nodegroup NG2 -node n5

   bind cluster nodegroup NG2 -node n6

4. Create another spare nodegroup and bind the requisite nodes.
   add cluster nodegroup NG3 -state SPARE -priority 10

   bind cluster nodegroup NG3 -node n7

   bind cluster nodegroup NG3 -node n8

   bind cluster nodegroup NG3 -node n9

Synchronizing Cluster Configurations

Feb 13, 2015

NetScaler configurations that are available on the configuration coordinator are synchronized to the other nodes of the cluster when:

- A node joins the cluster
- A node rejoins the cluster
- A new command is executed through the cluster IP address.

Additionally, you can forcefully synchronize the configurations that are available on the configuration coordinator (full synchronization) to a specific cluster node. Make sure you synchronize one cluster node at a time, otherwise the cluster can get affected.

To synchronize cluster configurations by using the command line interface

At the command prompt of the appliance on which you want to synchronize the configurations, type:

force cluster sync

To synchronize cluster configurations by using the configuration utility

1. Log on to the appliance on which you want to synchronize the configurations.
2. Navigate to System > Cluster.
3. In the details pane, under Utilities, click Force cluster sync.
4. Click OK.

Synchronizing Time Across Cluster Nodes

Mar 31, 2015

The cluster uses Precision Time Protocol (PTP) to synchronize the time across cluster nodes. PTP uses multicast packets to synchronize the time. If there are some issues in time synchronization, you must disable PTP and configure Network Time Protocol (NTP) on the cluster.

To enable/disable PTP by using the command line interface

At the command prompt of the cluster IP address, type:

set ptp -state disable

To enable/disable PTP by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster.
3. In the details pane, under Utilities, click Configure PTP Settings.
4. In the Enable/Disable PTP dialog box, select whether you want to enable or disable PTP.
5. Click OK.

Synchronizing Cluster Files

Feb 13, 2015

The files available on the configuration coordinator are called cluster files. These files are automatically synchronized on the other cluster nodes when the node is added to the cluster and periodically, during the lifetime of the cluster. Additionally, you can manually synchronize the cluster files.

The directories and files from the configuration coordinator that are synchronized are:

- /nsconfig/ssl/
- /var/netscaler/ssl/
- /var/vpn/bookmark/
- /nsconfig/dns/
- /nsconfig/htmlinjection/
- /netscaler/htmlinjection/ens/
- /nsconfig/monitors/
- /nsconfig/nstemplates/
- /nsconfig/ssh/
- /nsconfig/rc.netscaler
- /nsconfig/resolv.conf
- /nsconfig/inetd.conf
- /nsconfig/syslog.conf
- /nsconfig/snmpd.conf
- /nsconfig/ntp.conf
- /nsconfig/httpd.conf
- /nsconfig/sshd_config
- /nsconfig/hosts
- /nsconfig/enckey
- /var/nslw.bin/etc/krb5.conf
- /var/nslw.bin/etc/krb5.keytab
- /var/lib/likewise/db/
- /var/download/
- /var/wi/tomcat/webapps/
- /var/wi/tomcat/conf/Catalina/localhost/
- /var/wi/java_home/lib/security/cacerts
- /var/wi/java_home/jre/lib/security/cacerts
- /nsconfig/license/
- /nsconfig/rc.conf

To synchronize cluster files by using the command line interface

At the command prompt of the cluster IP address, type:

sync cluster files <mode>

To synchronize cluster files by using the configuration utility

1. Log on to the cluster IP address.

2. Navigate to System > Cluster.
3. In the details pane, under Utilities, click Synchronize cluster files.
4. In the Synchronize cluster files dialog box, select the files to be synchronized in the Mode drop-down box.
5. Click OK.

Viewing the Statistics of a Cluster

Feb 13, 2015

You can view the statistics of a cluster instance and cluster nodes to evaluate the performance or to troubleshoot the operation of the cluster.

To view the statistics of a cluster instance by using the command line interface

At the command prompt of the cluster IP address, type:

stat cluster instance <clId>

To view the statistics of a cluster node by using the command line interface

At the command prompt of the cluster IP address, type:

stat cluster node <nodeid>

Note: When executed from the cluster IP address, this command displays the cluster level statistics. However, when executed from the NSIP address of a cluster node, the command displays node level statistics.

To view the statistics of a cluster instance by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster.
3. In the details pane, in the center of the page, click Statistics.

To view the statistics of a cluster node by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster > Nodes.
3. In the details pane, select a node and click Statistics to view the statistics of the node. To view the statistics of all the nodes, click Statistics without selecting a specific node.

Discovering NetScaler Appliances

Feb 13, 2015

You can discover NetScaler appliances present in the same subnet as the NSIP address of the configuration coordinator. The discovered appliances can then be added to the cluster.

Note: This operation is available only through the configuration utility.

To discover appliances by using the configuration utility

1. Log on to the cluster IP address.
2. Navigate to System > Cluster > Nodes.
3. In the details pane, at the bottom of the page, click Discover NetScalers.
4. In the Discover NetScalers dialog box, set the following parameters:
   - IP address range - Specify the range of IP addresses within which you want to discover appliances. For example, you can search for all NSIP addresses between 10.102.29.4 to 10.102.29.15 by specifying this option as 10.102.29.4 - 15.
   - Backplane interface - Specify the interfaces to be used as the backplane interface. This is an optional parameter. If you do not specify this parameter, you must update it after the node is added to the cluster.
5. Click OK.
6. Select the appliances that you want to add to the cluster.
7. Click OK.

Disabling a Cluster Node

Feb 27, 2015

You can temporarily remove a node from a cluster by disabling the cluster instance on that node. A disabled node is not synchronized with the cluster configurations. When the node is enabled again, the cluster configurations are automatically synchronized on it. For more information, see Cluster Synchronization.

A disabled node cannot serve traffic and all existing connections on this node are terminated.

Note: If the configurations of a disabled non-configuration coordinator node are modified (through the NSIP address of the node), the configurations are not automatically synchronized on that node. You must manually synchronize the configurations as described in Synchronizing Cluster Configurations.

To disable a cluster node by using the command line interface

At the command prompt of the node that you want to disable, type:

disable cluster instance <clId>

Note: To disable the cluster, run the disable cluster instance command on the cluster IP address.

To disable a cluster node by using the configuration utility

1. On the node that you want to disable, navigate to System > Cluster, and click Manage Cluster.
2. In the Configure cluster instance dialog box, unselect the Enable cluster instance check box.
   Note: To disable the cluster instance on all the nodes, perform the above procedure on the cluster IP address.

Removing a Cluster Node

Feb 27, 2015

When a node is removed from the cluster, the cluster configurations are cleared from the node (by internally executing the clear ns config -extended command). The SNIP addresses and all VLAN configurations (except the default VLAN and NSVLAN) are also cleared from the appliance.

Note:

- If the deleted node was the cluster configuration coordinator, another node is automatically selected as the cluster configuration coordinator, and the cluster IP address is assigned to that node. All the current cluster IP address sessions will be invalid and you will have to start a new session.
- To delete the whole cluster, you must remove each node individually. When you remove the last node, the cluster IP address(es) are deleted.
- When an active node is removed, the traffic serving capability of the cluster is reduced by one node. Existing connections on this node are terminated.

For NetScaler 10.1 and later versions

Log on to the cluster IP address and at the command prompt, type:

rm cluster node <nodeId>

Note: If the cluster IP address is unreachable from the node, execute the rm cluster instance command on the NSIP address of that node itself.

For NetScaler 10

1. Log on to the node that you want to remove from the cluster and remove the reference to the cluster instance.
   rm cluster instance <clId>

   save ns config

2. Log on to the cluster IP address and remove the node from which you removed the cluster instance.
   rm cluster node <nodeId>

   save ns config

   Make sure you do not run the rm cluster node command from the local node as this results in inconsistent configurations between the configuration coordinator and the node.

On the cluster IP address, navigate to System > Cluster > Nodes, select the node you want to remove and click Remove.

Removing a Node from a Cluster Deployed Using Cluster Link Aggregation

Feb 13, 2015

To remove a node from a cluster that uses cluster link aggregation as the traffic distribution mechanism, you must make sure that the node is made passive so that it does not receive any traffic and then, on the upstream switch, remove the corresponding interface from the channel.

For detailed information on cluster link aggregation, see Using Cluster Link Aggregation.

1. Log on to the cluster IP address.
2. Set the state of the cluster node that you want to remove to PASSIVE.
   set cluster node <nodeId> -state PASSIVE
3. On the upstream switch, remove the corresponding interface from the channel by using switch-specific commands.
   Note: You do not have to manually remove the nodes interface on the cluster link aggregation channel. It is automatically removed when the node is deleted in the next step.
4. Remove the node from the cluster.
   rm cluster node <nodeId>

Cluster Setup and Usage Scenarios

Mar 31, 2015

This section aims at explaining some scenarios in which the NetScaler cluster can be setup and also how it can be configured for different features and network topologies. These are just some scenarios that we have documented. Provide feedback if you want some other scenarios to be included.

Creating a Two-Node Cluster

Feb 17, 2015

A two-node cluster is an exception to the rule that a cluster is functional only when a minimum of (n/2 +1) nodes, where n is the number of cluster nodes, are able to serve traffic. If that formula were applied to a two-node cluster, the cluster would fail if one node went down (n/2 +1=2).

A two-node cluster is functional even if only one node is able to serve traffic.

Creating a two node cluster is the same as creating any other cluster. You must add one node as the configuration coordinator and the other node as the other cluster node.

Note: Incremental configuration synchronization is not supported in a two-node cluster. Only full synchronization is supported.

Migrating an HA Setup to a Cluster Setup

Feb 17, 2015

An existing high availability (HA) setup can be migrated to a cluster setup by first removing the appliances from the HA setup and then creating the NetScaler cluster. This approach will result in a downtime for the application.

Consider an HA setup with appliances NS0 (10.102.97.131) and NS1 (10.102.97.132). NS0 is the primary and NS1 is the secondary appliance of the HA setup.

1. Log on to each HA node and remove it from the HA setup.
   rm HA node <id>

   Example
   rm HA node 1
2. Go to the shell on one of the HA nodes and copy the ns.conf file to another .conf file (for example, ns_backup.conf).
3. On both the nodes, identify the network interfaces to be used for the cluster backplane. Make sure to configure the backplane switch appropriately.
4. Create the cluster on one of the appliances (for example, 10.102.97.131).
   ```
   //On the NSIP address of the first appliance
   add cluster instance 1
   add cluster node 0 10.102.97.131 -state ACTIVE -backplane 0/1/1
   add ns ip 10.102.97.133 255.255.255.255 -type CLIP
   enable cluster instance 1
   save ns config
   reboot -warm
   ```
5. Add the other appliance to the cluster.
   ```
   //On the cluster IP address
   add cluster node 1 10.102.97.132 -state ACTIVE -backplane 1/1/1

   //On the NSIP address of the appliance
   join cluster -clip 10.102.97.133 -password nsroot
   save ns config
   reboot -warm
   ```
6. After the two nodes are up and active, log on to the cluster IP address and modify the backed-up configuration file as follows:
   1. Remove the features that are not supported on a cluster. For the list of unsupported features, see NetScaler Features Supported by a Cluster. This is an optional step. If you do not perform this step, the execution of unsupported commands will fail.
   2. Remove the configurations that have interfaces, or update the interface names from the c/u convention to the n/c/u convention.
      Example

      ```
      add vlan 10 -ifnum 0/1
      ```

      should be changed to

add vlan 10 -ifnum 0/0/1 1/0/1

3. The backup configuration file can have SNIP addresses or MIP addresses. These addresses are striped on all the cluster nodes. It is recommended that you add spotted IP addresses for each node.
   Example

   add ns ip 1.1.1.1 255.255.255.0 -ownerNode 0
   add ns ip 1.1.1.2 255.255.255.0 -ownerNode 1
4. Update the hostname to specify the owner node.
   Example

   set ns hostname ns0 -ownerNode 0
   set ns hostname ns1 -ownerNode 1
5. Change all other relevant networking configuration that depend on spotted IPs. For example, L3 VLAN, RNAT configuration which uses SNIPs as NATIP, INAT rules that refers to SNIPs/MIPs).

7. Apply configurations from the backup configuration file to the configuration coordinator through the cluster IP address.
   batch -fileName <input_filename>

   Example
   batch -f ns_backup.conf
8. Configure appropriate client traffic distribution mechanism (ECMP, cluster LA or linksets).
9. Save the configuration.
   save ns config

The appliances of the HA setup are migrated to a cluster setup.

Migrating an HA Setup to a Cluster Setup without Downtime

Feb 17, 2015

An existing high availability (HA) setup can be migrated to a cluster setup by first removing the secondary appliance from the HA setup and using that appliance to create a single-node cluster. Then, after the cluster becomes operational and serves traffic, the primary appliance of the HA setup is added to the cluster. This approach will not result in a downtime for the application.

Consider an HA setup with appliances NS0 (10.102.97.131) and NS1 (10.102.97.132). NS0 is the primary and NS1 is the secondary appliance of the HA setup.

1. Go to the shell on one of the HA nodes and copy the ns.conf file to another .conf file (for example, ns_backup.conf).
   Note: Make sure HA pair is stable with respect to configurations.
2. Log on to the secondary appliance NS1 and clear all the configurations. This removes the secondary appliance from the HA setup and makes it a standalone appliance.
   ```
   clear ns config full
   ```
   Note:
   - The configurations are cleared to make sure that NS1 does not start owning the VIPs once it becomes a standalone appliance.
   - At this stage, NS0 is still active and continues to serve traffic.
3. Create a cluster on appliance NS1 and configure it as a PASSIVE node.
   ```
   //On the NSIP address of node NS1
   add cluster instance 1
   add cluster node 0 10.102.97.131 -state PASSIVE -backplane 0/1/1
   add ns ip 10.102.97.133 255.255.255.255 -type CLIP
   enable cluster instance 1
   save ns config
   reboot -warm
   ```
4. Modify the backed-up configuration file.
   1. Remove the features that are not supported on a cluster. For the list of unsupported features, see NetScaler Features Supported by a Cluster. This is an optional step. If you do not perform this step, the execution of unsupported commands will fail.
   2. Remove the configurations that have interfaces, or update the interface names from the c/u convention to the n/c/u convention.
      Example

      ```
      add vlan 10 -ifnum 0/1
      ```

      should be changed to

      ```
      add vlan 10 -ifnum 0/0/1 1/0/1
      ```
   3. The backup configuration file can have SNIP addresses or MIP addresses. These addresses are striped on all the

cluster nodes. It is recommended that you add spotted IP addresses for each node.

Example

```
add ns ip 1.1.1.1 255.255.255.0 -ownerNode 0
add ns ip 1.1.1.2 255.255.255.0 -ownerNode 1
```

4. Update the hostname to specify the owner node.

Example

```
set ns hostname ns0 -ownerNode 0
set ns hostname ns1 -ownerNode 1
```

5. Change all other relevant networking configuration that depend on spotted IPs. For example, L3 VLAN, RNAT configuration which uses SNIPs as NATIP, INAT rules that refers to SNIPs/MIPs).

5. On the cluster, do the following:
   1. Make the topological changes to the cluster by connecting the cluster backplane, the cluster link aggregation channel, and so on.
   2. Apply configurations from the backup configuration file to the configuration coordinator through the cluster IP address.
      ```
      batch -f ns_backup.conf
      ```
   3. Configure external traffic distribution mechanisms like ECMP or cluster link aggregation.
6. Switch-over the traffic from the HA setup to the single-node cluster setup.
   1. Disable all interfaces on the primary appliance NS0.
      ```
      disable interface <interface id>
      ```
   2. Configure the cluster node as an ACTIVE node.
      ```
      set cluster node 0 -state ACTIVE
      ```

   Note: There can be a small amount (in the order of seconds) of downtime between disabling the interfaces and making the cluster node active.
7. On the primary appliance NS0, do the following:
   1. Clear all the configurations.
      ```
      clear ns config full
      ```
   2. Enable all the interfaces.
      ```
      enable interface <interface id>
      ```
   3. Add the appliance to the cluster.
      ```
      //On the cluster IP address (in this sample, 10.102.97.133)
      add cluster node 1 10.102.97.132 -state PASSIVE -backplane 1/1/1

      //On the NSIP address of the appliance
      join cluster -clip 10.102.97.133 -password nsroot
      save ns config
      reboot -warm
      ```
   4. Perform the required topological and configuration changes.
   5. Configure NS0 as an ACTIVE node.
      ```
      set cluster node 1 -state ACTIVE
      ```

The appliances of the HA setup are migrated to a cluster setup without any downtime for the application.

Using Cache Redirection in a Cluster

Feb 17, 2015

Cache redirection in a cluster works in the same way as it does on a standalone NetScaler appliance. The only difference is that the configurations are done on the cluster IP address. For more information on cache redirection, see "Cache Redirection."

Points to remember when using cache redirection in transparent mode on a cluster:

- Before configuring cache redirection, make sure that you have connected all nodes to the external switch and that you have linksets configured. Otherwise, client requests will be dropped.
- When MAC mode is enabled on a load balancing virtual server, make sure MBF mode is enabled on the cluster by using the enable ns mode MBF command. Otherwise, the requests are sent to origin server directly instead of being sent to the cache server.

Using L2 Mode in a Cluster Setup

Oct 08, 2014

Note: Supported from NetScaler 10.5 and later releases.

To use L2 mode in a cluster setup, you must make sure of the following:

- Spotted IP addresses must be available on all the nodes as required.
- Linksets must be used to communicate with the external network.
- Asymmetric topologies or asymmetric cluster LA groups are not supported.
- Cluster LA group is recommended.
- Traffic is distributed between the cluster nodes only for deployments where services exist.

Using Cluster LA Channel with Linksets

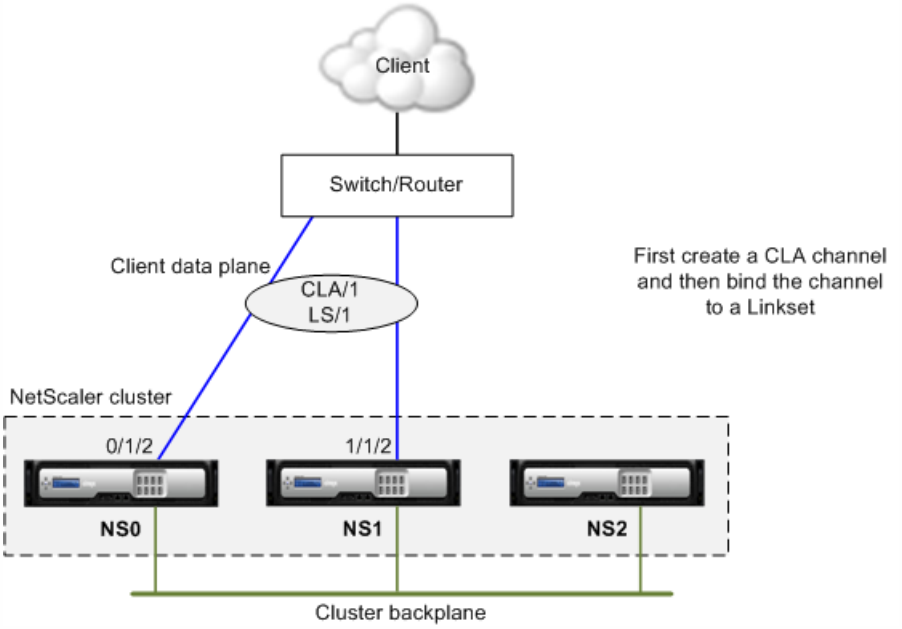
Feb 17, 2015

In an asymmetric cluster topology, some cluster nodes are not connected to the upstream network. In such a case, you must use linksets. To optimize the performance, you can bind the interfaces that are connected to the switch as a cluster LA channel and then bind the channel to a linkset.

To understand how a combination of cluster LA channel and linksets can be used, consider a three-node cluster for which the upstream switch has only two ports available. You can connect two of the cluster nodes to the switch and leave the other node unconnected.

Note: Similarly, you can also use a combination of ECMP and linksets in an asymmetric topology.

Figure 1. Linksets and cluster LA channel topology

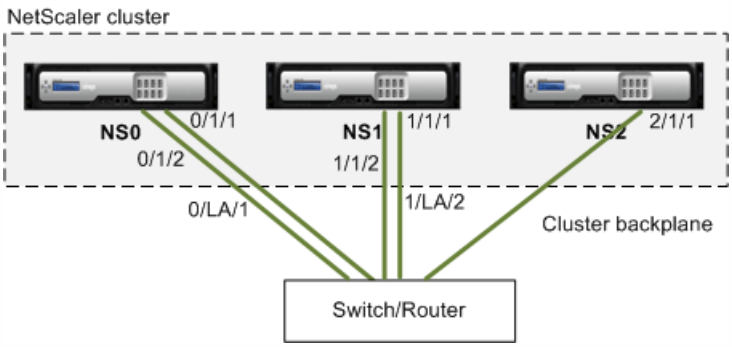


1. Log on to the cluster IP address.
2. Bind the connected interfaces to a cluster LA channel.
   ```
   add channel CLA/1 –ifnum 0/1/2 1/1/2
   ```
3. Bind the cluster LA channel to the linkset.
   ```
   add linkset LS/1 -ifnum CLA/1
   ```

Backplane on LA Channel

Feb 17, 2015

In this deployment, LA channels are used for the cluster backplane.



NS0 - nodeId: 0, NSIP: 10.102.29.60

NS1 - nodeId: 1, NSIP: 10.102.29.70

NS2 - nodeId: 2, NSIP: 10.102.29.80

To deploy a cluster with the backplane interfaces as LA channels

1. Create a cluster of nodes NS0, NS1, and NS2.
   1. Log on to the first node that you want to add to the cluster and do the following:
      ```
      create cluster instance 1
      add cluster node 0 10.102.29.60 -state ACTIVE
      enable cluster instance 1
      add ns ip 10.102.29.61 255.255.255.255 -type CLIP
      save ns config
      reboot -warm
      ```
   2. Log on to the cluster IP address and do the following:
      ```
      add cluster node 1 10.102.29.70 -state ACTIVE
      add cluster node 2 10.102.29.80 -state ACTIVE
      ```
   3. Log on to the nodes 10.102.29.70 and 10.102.29.80 to join the nodes to the cluster.
      ```
      join cluster -clip 10.102.29.61 -password nsroot
      save ns config
      reboot -warm
      ```

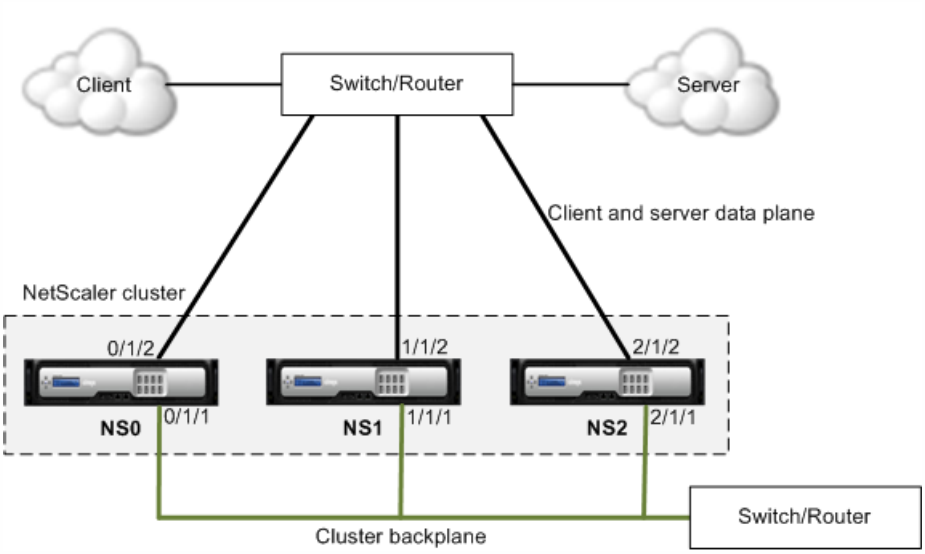
   As seen in the above commands the interfaces 0/1/1, 1/1/1, and 2/1/1 are configured as the backplane interfaces of the three cluster nodes.

2. Log on to the cluster IP address and do the following:
   1. Create the LA channels for nodes NS0 and NS1.
      ```
      add channel 0/LA/1 -ifnum 0/1/1 0/1/2
      add channel 1/LA/2 -ifnum 1/1/1 1/1/2
      ```
   2. Configure the backplane for the cluster nodes.
      ```
      set cluster node 0 -backplane 0/LA/1
      set cluster node 1 -backplane 1/LA/2
      set cluster node 2 -backplane 2/1/1
      ```

Common Interfaces for Client and Server and Dedicated Interfaces for Backplane

Feb 17, 2015

This is a one-arm deployment of the NetScaler cluster. In this deployment, the client and server networks use the same interfaces to communicate with the cluster. The cluster backplane uses dedicated interfaces for inter-node communication.



NS0 - nodeId: 0, NSIP: 10.102.29.60

NS1 - nodeId: 1, NSIP: 10.102.29.70

NS2 - nodeId: 2, NSIP: 10.102.29.80

To deploy a cluster with a common interface for the client and server and a different interface for the cluster backplane

1. Create a cluster of nodes NS0, NS1, and NS2.
    1. Log on to the first node that you want to add to the cluster and do the following:
       ```
       create cluster instance 1
       add cluster node 0 10.102.29.60 -state ACTIVE -backplane 0/1/1
       enable cluster instance 1
       add ns ip 10.102.29.61 255.255.255.255 -type CLIP
       save ns config
       reboot -warm
       ```
    2. Log on to the cluster IP address and do the following:
       ```
       add cluster node 1 10.102.29.70 -state ACTIVE -backplane 1/1/1
       add cluster node 2 10.102.29.80 -state ACTIVE -backplane 2/1/1
       ```
    3. Log on to the nodes 10.102.29.70 and 10.102.29.80 to join the nodes to the cluster.
       ```
       join cluster -clip 10.102.29.61 -password nsroot
       save ns config
       ```

```
reboot -warm
```

As seen in the above commands the interfaces 0/1/1, 1/1/1, and 2/1/1 are configured as the backplane interfaces of the three cluster nodes.

2. On the cluster IP address, create VLANs for the backplane interfaces and for the client and server interfaces.

```
//For the backplane interfaces
add vlan 10
bind vlan 10 0/1/1 1/1/1 2/1/1

//For the interfaces that are connected to the client and server networks.
add vlan 20
bind vlan 20 0/1/2 1/1/2 2/1/2
```

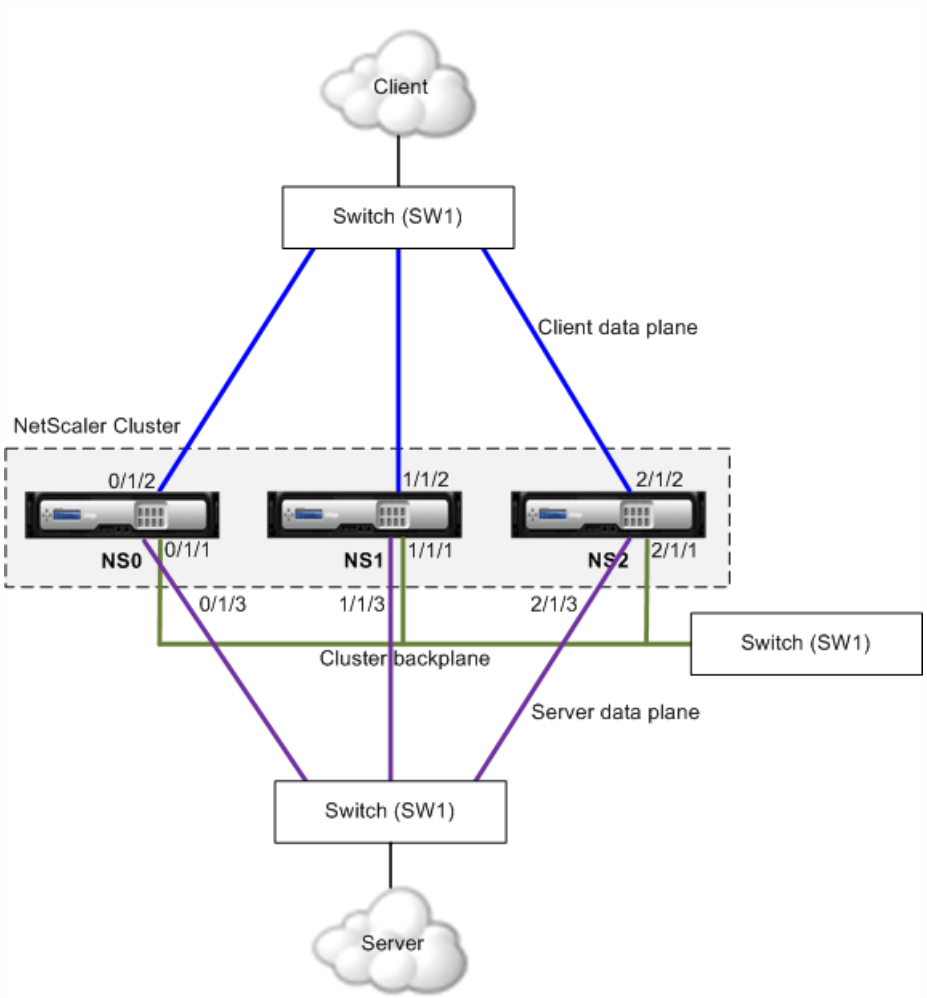
3. On the switch, create VLANs for the interfaces corresponding to the backplane interfaces and the client and server interfaces. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1) switch. Similar configurations must be performed on other switches.

```
//For the backplane interfaces. Repeat for each interface...
interface Ethernet2/47
switchport access vlan 100
switchport mode access
end

//For the interfaces connected to the client and server networks. Repeat for each interface...
interface Ethernet2/47
switchport access vlan 200
switchport mode access
end
```

Common Switch for Client, Server, and Backplane

Feb 17, 2015

In this deployment, the client, server, and backplane use dedicated interfaces on the same switch to communicate with the NetScaler cluster.



NS0 - nodeId: 0, NSIP: 10.102.29.60

NS1 - nodeId: 1, NSIP: 10.102.29.70

NS2 - nodeId: 2, NSIP: 10.102.29.80

To deploy a cluster with a common switch for the client, server, and backplane

1. Create a cluster of nodes NS0, NS1, and NS2.
   1. Log on to the first node that you want to add to the cluster and do the following:

      ```
      create cluster instance 1
      add cluster node 0 10.102.29.60 -state ACTIVE -backplane 0/1/1
      enable cluster instance 1
      add ns ip 10.102.29.61 255.255.255.255 -type CLIP
      save ns config
      ```

reboot -warm

2. Log on to the cluster IP address and do the following:
   add cluster node 1 10.102.29.70 -state ACTIVE -backplane 1/1/1
   add cluster node 2 10.102.29.80 -state ACTIVE -backplane 2/1/1
3. Log on to the nodes 10.102.29.70 and 10.102.29.80 to join the nodes to the cluster.
   join cluster -clip 10.102.29.61 -password nsroot
   save ns config
   reboot -warm

   As seen in the above commands the interfaces 0/1/1, 1/1/1, and 2/1/1 are configured as the backplane interfaces of the three cluster nodes.

2. On the cluster IP address, create VLANs for the backplane, client, and server interfaces.

```
//For the backplane interfaces
add vlan 10
bind vlan 10 0/1/1 1/1/1 2/1/1

//For the client-side interfaces
add vlan 20
bind vlan 20 0/1/2 1/1/2 2/1/2

//For the server-side interfaces
add vlan 30
bind vlan 30 0/1/3 1/1/3 2/1/3
```

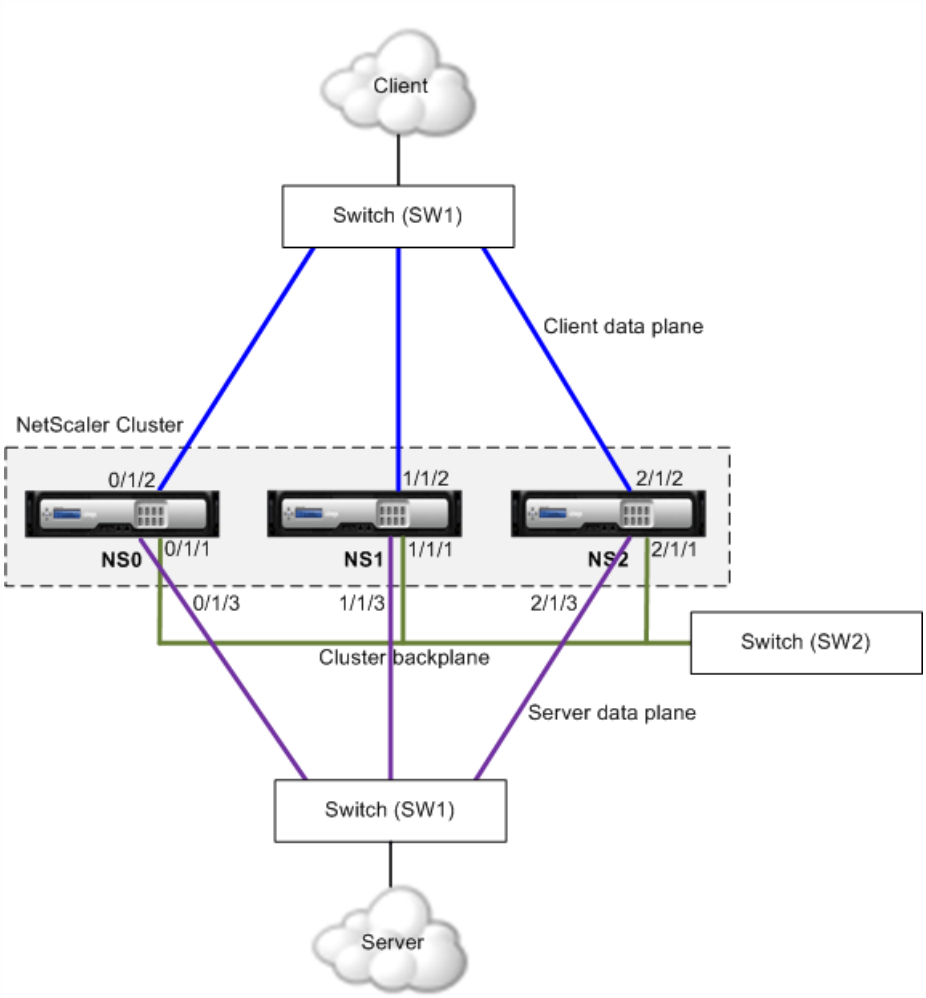
3. On the switch, create VLANs for the interfaces corresponding to the backplane interfaces and the client and server interfaces. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1) switch. Similar configurations must be performed on other switches.

```
//For the backplane interfaces. Repeat for each interface...
interface Ethernet2/47
switchport access vlan 100
switchport mode access
end

//For the client interfaces. Repeat for each interface...
interface Ethernet2/48
switchport access vlan 200
switchport mode access
end

//For the server interfaces. Repeat for each interface...
interface Ethernet2/49
switchport access vlan 300
switchport mode access
end
```

Common Switch for Client and Server and Dedicated Switch for Backplane

Feb 17, 2015

In this deployment, the clients and servers use different interfaces on the same switch to communicate with the NetScaler cluster. The cluster backplane uses a dedicated switch for inter-node communication.



NS0 - nodeId: 0, NSIP: 10.102.29.60

NS1 - nodeId: 1, NSIP: 10.102.29.70

NS2 - nodeId: 2, NSIP: 10.102.29.80

To deploy a cluster with the same switch for the clients and servers and a different switch for the cluster backplane

1. Create a cluster of nodes NS0, NS1, and NS2.
   1. Log on to the first node that you want to add to the cluster and do the following:
      ```
      create cluster instance 1
      add cluster node 0 10.102.29.60 -state ACTIVE -backplane 0/1/1
      ```

```
enable cluster instance 1
add ns ip 10.102.29.61 255.255.255.255 -type CLIP
save ns config
reboot -warm
```

2. Log on to the cluster IP address and do the following:

```
add cluster node 1 10.102.29.70 -state ACTIVE -backplane 1/1/1
add cluster node 2 10.102.29.80 -state ACTIVE -backplane 2/1/1
```

3. Log on to the nodes 10.102.29.70 and 10.102.29.80 to join the nodes to the cluster.

```
join cluster -clip 10.102.29.61 -password nsroot
save ns config
reboot -warm
```

As seen in the above commands the interfaces 0/1/1, 1/1/1, and 2/1/1 are configured as the backplane interfaces of the three cluster nodes.

2. On the cluster IP address, create VLANs for the backplane, client, and server interfaces.

```
//For the backplane interfaces
add vlan 10
bind vlan 10 0/1/1 1/1/1 2/1/1

//For the client-side interfaces
add vlan 20
bind vlan 20 0/1/2 1/1/2 2/1/2

//For the server-side interfaces
add vlan 30
bind vlan 30 0/1/3 1/1/3 2/1/3
```

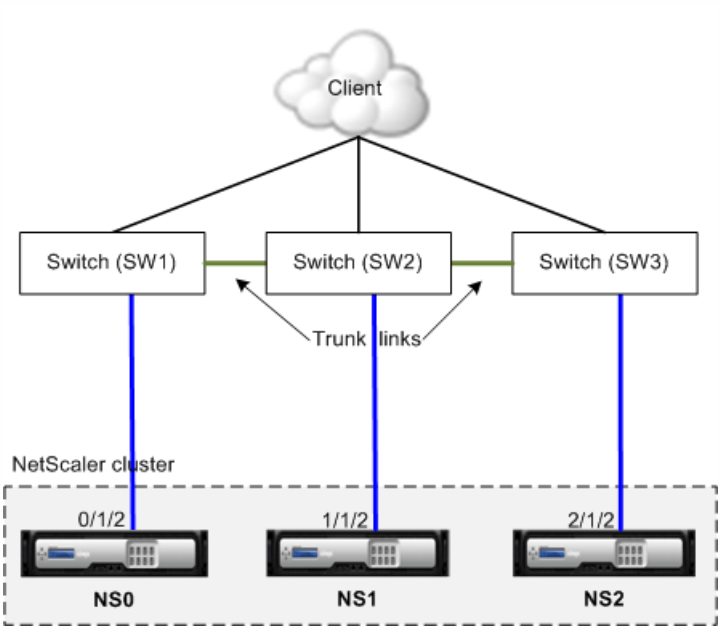
3. On the switch, create VLANs for the interfaces corresponding to the backplane interfaces and the client and server interfaces. The following sample configurations are provided for the Cisco® Nexus 7000 C7010 Release 5.2(1) switch. Similar configurations must be performed on other switches.

```
//For the backplane interfaces. Repeat for each interface...
interface Ethernet2/47
switchport access vlan 100
switchport mode access
end

//For the client interfaces. Repeat for each interface...
interface Ethernet2/48
switchport access vlan 200
switchport mode access
end

//For the server interfaces. Repeat for each interface...
interface Ethernet2/49
switchport access vlan 300
switchport mode access
end
```

Different Switch for Every Node

Feb 17, 2015

In this deployment, each cluster node is connected to a different switch and trunk links are configured between the switches.



The cluster configurations will be the same as the other deployments scenarios. Most of the client-side configurations will be done on the client-side switches.

Sample Cluster Configurations

Feb 09, 2015

The following example can be used to configure a four-node cluster with ECMP, cluster LA, or Linksets.

1. Create the cluster.
   1. Log on to first node.
   2. Add the cluster instance.
      ```
      add cluster instance 1
      ```
   3. Add the first node to the cluster.
      ```
      add cluster node 0 10.102.33.184 -backplane 0/1/1
      ```
   4. Enable the cluster instance.
      ```
      enable cluster instance 1
      ```
   5. Add the cluster IP address.
      ```
      add ns ip 10.102.33.185 255.255.255.255 -type CLIP
      ```
   6. Save the configurations.
      ```
      save ns config
      ```
   7. Warm reboot the appliance.
      ```
      reboot -warm
      ```
2. Add the other three nodes to the cluster.
   1. Log on to cluster IP address.
   2. Add the second node to the cluster.
      ```
      add cluster node 1 10.102.33.187 -backplane 1/1/1
      ```
   3. Add the third node to the cluster.
      ```
      add cluster node 2 10.102.33.188 -backplane 2/1/1
      ```
   4. Add the fourth node to the cluster.
      ```
      add cluster node 3 10.102.33.189 -backplane 3/1/1
      ```
3. Join the added nodes to the cluster. This step is not applicable for the first node.
   1. Log on to each newly added node.
   2. Join the node to the cluster.
      ```
      join cluster -clip 10.102.33.185 -password nsroot
      ```
   3. Save the configuration.
      ```
      save ns config
      ```
   4. Warm reboot the appliance.
      ```
      reboot -warm
      ```
4. Configure the NetScaler cluster through the cluster IP address.
   ```
   // Enable load balancing feature
   enable ns feature lb

   // Add a load balancing virtual server
   add lb vserver first_lbvserver http
   ....
   ....
   ```
5. Configure any one of the following (ECMP, cluster LA, or Linkset) traffic distribution mechanisms for the cluster.
   - ECMP
     1. Log on to the cluster IP address.

2. Enable the OSPF routing protocol.
   enable ns feature ospf
3. Add a VLAN.
   add vlan 97
4. Bind the interfaces of the cluster nodes to the VLAN.
   bind vlan 97 -ifnum 0/1/4 1/1/4 2/1/4 3/1/4
5. Add a spotted SNIP on each node and enable dynamic routing on it.
   add ns ip 1.1.1.10 255.255.255.0 -ownerNode 0 -dynamicRouting ENABLED
   add ns ip 1.1.1.11 255.255.255.0 -ownerNode 1 -dynamicRouting ENABLED
   add ns ip 1.1.1.12 255.255.255.0 -ownerNode 2 -dynamicRouting ENABLED
   add ns ip 1.1.1.13 255.255.255.0 -ownerNode 3 -dynamicRouting ENABLED
6. Bind one of the SNIP addresses to the VLAN.
   bind vlan 97 -ipAddress 1.1.1.10 255.255.255.0
7. Configure the routing protocol on ZebOS by using vtysh shell.

- **Static cluster LA**

  1. Log on to the cluster IP address.
  2. Add a cluster LA channel.
     add channel CLA/1 -speed 1000
  3. Bind the interfaces to the cluster LA channel.
     bind channel CLA/1 0/1/5 1/1/5 2/1/5 3/1/5
  4. Perform equivalent configuration on the switch.

- **Dynamic cluster LA**

  1. Log on to the cluster IP address.
  2. Add the interfaces to the cluster LA channel.
     set interface 0/1/5 -lacpmode active -lacpkey 5 -lagtype cluster
     set interface 1/1/5 -lacpmode active -lacpkey 5 -lagtype cluster
     set interface 2/1/5 -lacpmode active -lacpkey 5 -lagtype cluster
     set interface 3/1/5 -lacpmode active -lacpkey 5 -lagtype cluster
  3. Perform equivalent configuration on the switch.

- **Linksets.** Assume that the node with nodeId 3 is not connected to the switch. You must configure a linkset so that the unconnected node can use the other node interfaces to communicate with the switch.

  1. Log on to the cluster IP address.
  2. Add a linkset.
     add linkset LS/1
  3. Bind the connected interfaces to the linkset.
     bind linkset LS/1 -ifnum 0/1/6 1/1/6 2/1/6

6. Update the state of the cluster nodes to ACTIVE.
   set cluster node 0 -state ACTIVE
   set cluster node 1 -state ACTIVE
   set cluster node 2 -state ACTIVE
   set cluster node 3 -state ACTIVE

Upgrading or Downgrading the NetScaler Cluster

May 20, 2015

All the nodes of a NetScaler cluster must be running the same software version. Therefore, to upgrade or downgrade the cluster, you must upgrade or downgrade each NetScaler appliance of the cluster, one node at a time.

A node that is being upgraded or downgraded is not removed from the cluster. The node continues to be a part of the cluster and serves traffic uninterrupted, except for the down-time when the node reboots after it is upgraded or downgraded. However, due to software version mismatch among the cluster nodes, configuration propagation is disabled on the cluster and is enabled only after all the cluster nodes are of the same version. Since configuration propagation is disabled during upgrading on downgrading a cluster, you cannot perform any configurations through the cluster IP address during this time.

Points to note before upgrading or downgrading the cluster

- You cannot add cluster nodes while upgrading or downgrading the cluster software version.
- You can perform node-level configurations through the NSIP address of individual nodes, but you must make sure that you perform the same configurations on all the nodes to maintain them in synch.
- You cannot execute the start nstrace command from the cluster IP address when the cluster is being upgraded. However, you can get the trace of individual nodes by performing this operation on individual cluster nodes through their NetScaler IP (NSIP) address.
- Configurations can be lost during the downgrade of the cluster.
- Owing to changes in cluster licensing that were made in NetScaler 10.5 Build 52.11 (see license requirements), look into the following:
  - If the cluster is setup in a build prior to NetScaler 10.5 Build 52.11, the cluster will work with the separate cluster license file. No changes are required.
  - If the cluster is setup in NetScaler 10.5 Build 52.11 or later releases and then downgraded to a build prior to NetScaler 10.5 Build 52.11, the downgraded cluster will not work as it now expects a separate cluster license file.
- While upgrading from any NetScaler 10.1 build to a later release, syncookie must be disabled on all TCP profiles (using the "set ns tcpProfile <name> -synCookie DISABLED" command) and after that a striped SNIP must be added on the CLIP subnet. Once upgraded, syncookie can be enabled again.
- While upgrading the NetScaler appliance from a NetScaler 10.1 build to a NetScaler 10.5 build, do not execute the "show audit messages" command as this can cause the NetScaler appliance to crash.
- NetScaler 10.5 54.x and 55.x builds are not suitable for cluster deployment. This is because, for services that need probing, SYN packets are processed locally (on the flow receiver) even though syncookie is disabled.
- When a cluster is being upgraded, it is possible that the upgraded nodes have some additional features activated that are not available on the nodes that are not upgraded. This results in a license mismatch warning while the cluster is being upgraded. This warning will be automatically resolved when all the cluster nodes are upgraded.

To upgrade or downgrade the software of the cluster nodes

1. Make sure the cluster is stable and the configurations are synchronized on all the nodes.
2. For each cluster node perform the following:
   Note: Citrix recommends that you wait for the previous node to become active before upgrading or downgrading the next node.
   1. Upgrade or downgrade the cluster node. For detailed information about upgrading and downgrading the software of an appliance, see "Upgrading or Downgrading the System Software".
   2. Save the configurations.

3. Reboot the appliance.

3. Repeat step 2 for each of the other cluster nodes.

Operations Supported on Individual Cluster Nodes

Mar 20, 2015

As a rule, NetScaler appliances that are a part of a cluster cannot be individually configured from their NSIP address. However, there are some operations that are an exception to this rule. These operations, when executed from the NSIP address, are not propagated to other cluster nodes.

The operations are:

- cluster instance (set | rm | enable | disable)
- cluster node (set | rm)
- nstrace (start | show | stop)
- interface (set | enable | disable)
- force cluster sync
- sync cluster files
- disable ntp sync
- save ns config
- reboot
- shutdown

For example, when you execute the command disable interface 1/1/1 from the NSIP address of a cluster node, the interface is disabled only on that node. Since the command is not propagated, the interface 1/1/1 remains enabled on all the other cluster nodes.

FAQs

Feb 04, 2016

Click here for clustering FAQs for NetScaler versions 11.0, 10.5, and 10.1.

Troubleshooting the NetScaler Cluster

Feb 09, 2015

If a failure occurs in a NetScaler cluster, the first step in troubleshooting is to get information on the cluster instance and the cluster nodes by running the show cluster instance <clId> and show cluster node <nodeId> commands respectively.

If you are not able to find the issue by using the above two approaches, you can use one of the following:

- **Isolate the source of the failure.** Try bypassing the cluster to reach the server. If the attempt is successful, the problem is probably with the cluster setup.
- **Check the commands recently executed.** Run the history command to check the recent configurations performed on the cluster. You can also review the ns.conf file to verify the configurations that have been implemented.
- **Check the ns.log files.** Use the log files, available in the /var/log/ directory of each node, to identify the commands executed, status of commands, and the state changes.
- **Check the newnslog files.** Use the newnslog files, available in the /var/nslog/ directory of each node, to identify the events that have occurred on the cluster nodes. You can view multiple newnslog files as a single file, by copying the files to a single directory, and then running the following command:
  nsconmsg -K newnslog-node<id> -K newnslog.node<id> -d current

If you still cannot resolve the issue, you can try tracing the packets on the cluster or use the show techsupport -scope cluster command to send the report to the technical support team.

Tracing the Packets of a NetScaler Cluster

Feb 09, 2015

The NetScaler operating system provides a utility called *nstrace* to get a dump of the packets that are received and sent out by an appliance. The utility stores the packets in trace files. You can use these files to debug problems in the flow of packets to the cluster nodes. The trace files must be viewed with the Wireshark application.

Some salient aspects of the nstrace utility are:

- Can be configured to trace packets selectively by using classic expressions and default expressions.
- Can capture the trace in multiple formats: nstrace format (.cap) and TCP dump format (.pcap).
- Can aggregate the trace files of all cluster nodes on the configuration coordinator.
- Can merge multiple trace files into a single trace file (only for .cap files).

You can use the nstrace utility from the NetScaler command line or the NetScaler shell.

Run the start nstrace command on the appliance. The command creates trace files in the /var/nstrace/<date-timestamp> directory. The trace file names are of the form nstrace<id>.cap.

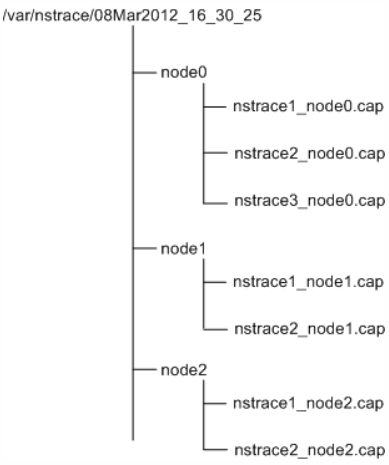
You can view the status by executing the show nstrace command. You can stop tracing the packets by executing the stop nstrace command.

Note: You can also run the nstrace utility from the NetScaler shell by executing the nstrace.sh file. However, it is recommended that you use the nstrace utility through the NetScaler command line interface.

You can trace the packets on all the cluster nodes and obtain all the trace files on the configuration coordinator.

Run the start nstrace command on the cluster IP address. The command is propagated and executed on all the cluster nodes. The trace files are stored in individual cluster nodes in the /var/nstrace/<date-timestamp> directory. The trace file names are of the form nstrace<id>_node<id>.cap.

You can use the trace files of each node to debug the nodes operations. But if you want the trace files of all cluster nodes in one location, you must run the stop nstrace command on the cluster IP address. The trace files of all the nodes are downloaded on the cluster configuration coordinator in the /var/nstrace/<date-timestamp> directory as follows:



You can prepare a single file from the trace files (supported only for .cap files) obtained from the cluster nodes. The single trace files gives you a cumulative view of the trace of the cluster packets. The trace entries in the single trace file are sorted based on the time the packets were received on the cluster.

To merge the trace files, at the NetScaler shell, type:

nstracemerge.sh -srcdir <DIR> -dstdir <DIR> -filename <name> -filesize <num>

where,

- srcdir is the directory from which the trace files are merged. All trace files within this directory are merged into a single file.
- dstdir is the directory where the merged trace file are created.

- filename is the name of the trace file that is created.
- filesize is the size of the trace file.

Following are some examples of using the nstrace utility to filter packets.

- To trace the packets on the backplane interfaces of three nodes:
  Using classic expressions:

  start nstrace -filter "INTF == 0/1/1 && INTF == 1/1/1 && INTF == 2/1/1"
  Using default expressions:

  start nstrace -filter "CONNECTION.INTF.EQ("0/1/1") && CONNECTION.INTF.EQ("1/1/1") && CONNECTION.INTF.EQ("2/1/1")"
- To trace the packets from a source IP address 10.102.34.201 or from a system whose source port is greater than 80 and the service name is not "s1":
  Using classic expressions

  start nstrace -filter "SOURCEIP == 10.102.34.201 || (SVCNAME != s1 && SOURCEPORT > 80)"
  Using default expressions

  start nstrace -filter "CONNECTION.SRCIP.EQ(10.102.34.201) || (CONNECTION.SVCNAME.NE("s1") && CONNECTION.SRCPORT.GT(80))"

Troubleshooting Common Issues

Feb 09, 2015

While joining a node to the cluster, I get the following message, "ERROR: Invalid interface name/number." What must I do to resolve this error?

This error occurs if you provided an invalid or incorrect backplane interface while using the add cluster node command to add the node. To resolve this error, verify the interface you provided while adding the node. Make sure that you have not specified the appliance's management interface as the backplane interface, and that the <nodeId> bit of the interface is the same as the node's Id. For example, if the nodeId is 3, the backplane interface must be 3/<c>/<u>.

While joining a node to the cluster, I get the following message, "ERROR: Clustering cannot be enabled, because the local node is not a member of the cluster." What must I do to resolve this error?

This error occurs when you try to join a node without adding the node's NSIP to the cluster. To resolve this error, you must first add the node's NSIP address to the cluster by using the add cluster node command and then execute the join cluster command.

While joining a node to the cluster, I get the following message, "ERROR: Connection refused." What must I do to resolve this error?

This error can occur due to the following reasons:

- **Connectivity problems.** The node cannot connect to the cluster IP address. Try pinging the cluster IP address from the node that you are trying to join.
- **Duplicate cluster IP address.** Check to see if the cluster IP address exists on some non-cluster node. If it does, create a new cluster IP address and try re-joining the cluster.

While joining a node to the cluster, I get the following message, "ERROR: License mismatch between the configuration coordinator and the local node." What must I do to resolve this error?

The appliance that you are joining to the cluster must have the same licenses as the configuration coordinator. This error occurs when the licenses on the node you are joining do not match the licenses on the configuration coordinator. To resolve this error, run the following commands on both the nodes and compare the outputs.

From the command line:

- show ns hardware
- show ns license

From the shell:

- nsconmsg -g feature -d stats
- ls /nsconfig/license
- View the contents of the /var/log/license.log file

What must I do when the configurations of a cluster node are not in synch with the cluster configurations?

In most cases, the configurations are automatically synchronized between all the cluster nodes. However, if you feel that the configurations are not synchronized on a specific node, you must force the synchronization by executing the force cluster sync command from the node that you want to synchronize. For more information, see "Synchronizing Cluster

Configurations".

When configuring a cluster node, I get the following message, "ERROR: Session is read-only; connect to the cluster IP address to modify the configuration."

All configurations on a cluster must be done through the cluster IP address and the configurations are propagated to the other cluster nodes. All sessions established through the NetScaler IP (NSIP) address of individual nodes are read-only.

Why does the node state show "INACTIVE" when the node health shows "UP"?

A healthy node can be in the INACTIVE state for a number of reasons. A scan of ns.log or error counters can help you determine the exact reason.

How can I resolve the health of a node when its health shows "NOT UP"?

Node health "**Not UP**" indicates that there are some issues with the node. To know the root cause, you must run the show cluster node command. This command displays the node properties and the reason for the node failure.

What must I do when the health of a node shows as "NOT UP" and the reason indicates that configuration commands have failed on a node?

This issue arises when some commands are not executed on the cluster nodes. In such cases, you must make sure that the configurations are synchronized using one of the following options:

- If some of the cluster nodes are in this state, you must perform the force cluster synchronization operation on those nodes. For more information, see "Synchronizing Cluster Configurations".
- If all cluster nodes are in this state, you must disable and then enable the cluster instance on all the cluster nodes.

When I run the set vserver command, I get the following message, "No such resource." What must I do to resolve this issue?

The set vserver command is not supported in clustering. The unset vserver, enable vserver, disable vserver, and rm vserver commands are also not supported. However, the show vserver command is supported.

I cannot configure the cluster over a Telnet session. What must I do?

Over a telnet session, the cluster IP address can be accessed only in read-only mode. Therefore, you cannot configure a cluster over a telnet session.

I notice a significant time difference across the cluster nodes. What must I do to resolve this issue?

When PTP packets are dropped due to backplane switch or if the physical resources are over-committed in a virtual environment, the time will not get synchronized.
To synchronize the times, you must do the following on the cluster IP address:

1. Disable PTP.

   set ptp -state disable

2. Configure Network Time Protocol (NTP) for the cluster. For more information, see "Setting up Clock Synchronization".

What must I do, if there is no connectivity to the cluster IP address and the NSIP address of a cluster node?

If you cannot access to the cluster IP address or the NSIP of a cluster node, you must access the appliance through the serial console. For more information, see "Using the Command Line Interface".
If the NSIP address is reachable, you can SSH to the cluster IP address from the shell by executing the following command at the shell prompt:

```
# ssh nsroot@<cluster IP address>
```

What must I do to recover a cluster node that has connectivity issues?

To recover a node that has connectivity issues:

1. Disable the cluster instance on that node (since you cannot execute commands from the NSIP of a cluster node).
2. Execute the commands required to recover the node.
3. Enable the cluster instance on that node.

Some nodes of the cluster have two default routes. How can I remove the second default route from the cluster node?

To delete the additional default route, do the following on each node that has the extra route:

1. Disable the cluster instance.
   disable cluster instance <clId>
2. Remove the route.
   rm route <network> <netmask> <gateway>
3. Enable the cluster instance.
   enable cluster instance <clId>

The cluster functionality gets affected when an existing cluster node comes online. What must I do to resolve this issue?

If RPC password of node is changed from the cluster IP address when that node is out of the cluster, then, when the node comes online, there is a mismatch in rpc credentials and this could affect cluster functionality. To solve this issue, use the set ns rpcNode command to update the password on the NSIP of the node which has come online.

CloudBridge Connector

Oct 20, 2015

The CloudBridge Connector feature of the Citrix NetScaler appliance connects enterprise datacenters to external clouds and hosting environments, making the cloud a secure extension of your enterprise network. Cloud-hosted applications appear as though they are running on one contiguous enterprise network. With Citrix CloudBridge Connector, you can augment your datacenters with the capacity and efficiency available from cloud providers.

The CloudBridge Connector enables you to move your applications to the cloud to reduce costs and increase reliability.

In addition to using CloudBridge Connector between a datacenter and a cloud, you can use it to connect two datacenters for a high-capacity secure and accelerated link.

Understanding CloudBridge Connector

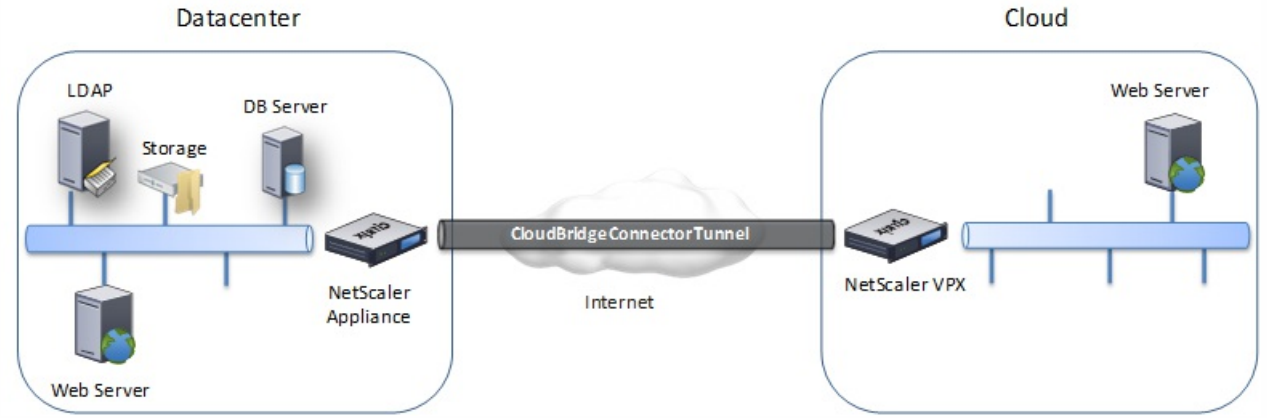
Updated: 2014-04-14

To implement the Citrix CloudBridge Connector solution, you connect a datacenter to another datacenter or an external cloud by setting up a tunnel called the CloudBridge Connector tunnel.

To connect a datacenter to another datacenter, you set up a CloudBridge Connector tunnel between two NetScaler appliances, one in each datacenter.

To connect a datacenter to an external cloud (for example, Amazon AWS cloud), you set up a CloudBridge Connector tunnel between a NetScaler appliance in the datacenter and a virtual appliance (VPX) that resides in the Cloud. The remote end point can be a CloudBridge Connector or a NetScaler VPX with platinum license.

The following illustration shows a CloudBridge Connector tunnel set up between a datacenter and an external cloud.



The appliances between which a CloudBridge Connector tunnel is set up are called the *end points* or *peers* of the CloudBridge Connector tunnel.

A CloudBridge Connector tunnel uses the following protocols:

- Generic Routing Encapsulation (GRE) protocol
- Open-standard IPSec Protocol suite, in transport mode

The GRE protocol provides a mechanism for encapsulating packets, from a wide variety of network protocols, to be

forwarded over another protocol. GRE is used to:

- Connect networks running non-IP and non-routable protocols.
- Bridge across a wide area network (WAN).
- Create a transport tunnel for any type of traffic that needs to be sent unchanged across a different network.

The GRE protocol encapsulates packets by adding a GRE header and a GRE IP header to the packets.

The Internet Protocol security (IPSec) protocol suite secures communication between peers in the CloudBridge Connector tunnel.

In a CloudBridge Connector tunnel, IPSec ensures:

- Data integrity
- Data origin authentication
- Data confidentiality (encryption)
- Protection against replay attacks

IPSec uses the transport mode in which the GRE encapsulated packet is encrypted. The encryption is done by the Encapsulating Security Payload (ESP) protocol. The ESP protocol ensures the integrity of the packet by using a HMAC hash function, and ensures confidentiality by using an encryption algorithm. After the packet is encrypted and the HMAC is calculated, an ESP header is generated. The ESP header is inserted after the GRE IP header and, an ESP trailer is inserted at the end of the encrypted payload.

Peers in the CloudBridge Connector tunnel use the Internet Key Exchange version (IKE) protocol (part of the IPSec protocol suite) to negotiate secure communication, as follows:

- The two peers mutually authenticate with each other, using one of the following authentication methods:
  - **Pre-shared key authentication**. A text string called a pre-shared key is manually configured on each peer. The pre-shared keys of the peers are matched against each other for authentication. Therefore, for the authentication to be successful, you must configure the same pre-shared key on each of the peers.
  - **Digital certificates authentication**. The initiator (sender) peer signs message interchange data by using its private key, and the other receiver peer uses the sender's public key to verify the signature. Typically, the public key is exchanged in messages containing an X.509v3 certificate. This certificate provides a level of assurance that a peer's identity as represented in the certificate is associated with a particular public key.
- The peers then negotiate to reach agreement on:
  - An encryption algorithm.
  - Cryptographic keys for encrypting data in one peer and decrypting the data in the other.

This agreement upon the security protocol, encryption algorithm and cryptographic keys is called a Security Association (SA). SAs are one-way (simplex). For example, when two peers, CB1 and CB2, are communicating through a Connector tunnel, CB1 has two Security Associations. One SA is used for processing out-bound packets, and the other SA is used for processing inbound packets.

SAs expire after a specified length of time, which is called the *lifetime*. The two peers use the Internet Key Exchange (IKE)

protocol (part of the IPSec protocol suite) to negotiate new cryptographic keys and establish new SAs. The purpose of the limited lifetime is to prevent attackers from cracking a key.

Configuring CloudBridge Connector Tunnel between two Datacenters

Aug 02, 2013

You can configure a CloudBridge Connector tunnel between two different datacenters to extend your network without reconfiguring it, and leverage the capabilities of the two datacenters. Having a CloudBridge Connector tunnel configured between the two geographically separated datacenters enables you to implement redundancy and safeguard your setup from failure. The CloudBridge Connector tunnel helps achieve optimal utilization of infrastructure and resources across two datacenters. The applications available across the two datacenters appear as local to the user.

To connect a datacenter to another datacenter, you set up a CloudBridge Connector tunnel between a NetScaler appliance that reside in one datacenter and another NetScaler appliance that reside in the other datacenter.

As an illustration of CloudBridge Connector tunnel between two different datacenters, consider an example in which a CloudBridge Connector tunnel is set up between NetScaler appliance NS_Appliance-1 in datacenter DC1 and NetScaler appliance NS_Appliance-2 in datacenter DC2.

Both NS_Appliance-1 and NS_Appliance-2 function in L2 and L3 mode. They enable communication between private networks in datacenters DC1 and DC2. In L3 mode, NS_Appliance-1 and NS_Appliance-2 enable communication between client CL1 in the datacenter DC1 and server S1 in the datacenter DC2 through the CloudBridge Connector tunnel. Client CL1 and server S1 are on different private networks.

Because client CL1 and server S1 are on different private networks, L3 mode is enabled on NS_Appliance-1 and NS_Appliance-2 and routes are updated as the following:

- CL1 have a route to NS_Appliance-1 for reaching S1
- NS_Appliance-1 have a route to NS_Appliance-2 for reaching S1
- S1 should have a route to NS_Appliance-2 for reaching CL1
- NS_Appliance-2 have a route to NS_Appliance-1 for reaching CL1

The following table lists the settings on NetScaler appliance NS_Appliance-1 in datacenter DC1.

Entity	Name	Details
The NSIP address		198.51.100.12
SNIP address		198.51.100.15
CloudBridge Connector tunnel	Cloud_Connector_DC1-DC2	• Local endpoint IP address of the CloudBridge Connector tunnel = 198.51.100.15 • Remote endpoint IP address of the CloudBridge Connector tunnel = 203.0.113.133

		GRE Tunnel Details • Name = Cloud_Connector_DC1-DC2 IPSec Profile Details • Name = Cloud_Connector_DC1-DC2 • Encryption algorithm = AES • Hash algorithm = HMAC SHA1

The following table lists the settings on NetScaler appliance NS_Appliance-2 in datacenter DC2.

Entity	Name	Details
The NSIP address		203.0.113.131
SNIP address		203.0.113.133
CloudBridge Connector tunnel	Cloud_Connector_DC1-DC2	• Local endpoint IP address of the CloudBridge Connector tunnel = 203.0.113.133 • Remote endpoint IP address of the CloudBridge Connector tunnel = 198.51.100.15 GRE Tunnel Details • Name = Cloud_Connector_DC1-DC2 IPSec Profile Details • Name = Cloud_Connector_DC1-DC2 • Encryption algorithm = AES • Hash algorithm = HMAC SHA1

Following is the traffic flow in the CloudBridge Connector tunnel:

1. Client CL1 sends a request to server S1.
2. The request reaches NetScaler appliance NS-Appliance-1.
3. NS_Appliance-1, checks its routing table and finds that the destination IP address of the request packet belongs to a subnet in datacenter DC2. The appliance decides to forward the packet to be sent across the CC-DC1-DC2 tunnel.
4. NS_Appliance-1 uses the GRE protocol to encapsulate each of the request packets by adding a GRE header and a GRE IP header to the packet. The GRE IP header has the destination IP address set to the IP address of the CloudBridge tunnel (CC-DC1-DC2) end point in DC2 side. This IP Address is a public SNIP address configured on the NetScaler instance running on the NetScaler appliance NS_Appliance-2.
5. For CloudBridge Connector tunnel CC-DC1-DC2, NS_Appliance-1 checks the stored IPSec security association (SA) parameters for processing outbound packets, as agreed between NS_Appliance-1 and NS_Appliance-2 . The IPSec Encapsulating Security Payload (ESP) protocol in NS_Appliance-1 uses these SA parameters for outbound packets, to encrypt the payload of the GRE encapsulated packet.
6. The ESP protocol ensures the packet's integrity and confidentiality by using the HMAC hash function and the encryption algorithm specified for the CloudBridge Connector tunnel CC-DC1-DC2 . The ESP protocol, after encrypting the GRE

payload and calculating the HMAC, generates an ESP header and an ESP trailer and inserts them before and at the end of the encrypted GRE payload, respectively.
7. The resulting packet is sent to NS_Appliance-2.
8. NS_Appliance-2 checks the stored IPSec security association (SA) parameters for processing inbound packets, as agreed between NS_Appliance-1 and NS_Appliance-2 for the CloudBridge Connector tunnel CC-DC1-DC2 . The IPSec ESP protocol on NS_Appliance-2 uses these SA parameters for inbound packets, and the ESP header of the request packet, to decrypt the packet.
9. NS_Appliance-2 then decapsulates the packet by removing the GRE header.
10. The resulting packet is the same packet as the one received by NS_Appliance-1 in step 2. This packet has the destination IP address set to the IP address of server S1. NS_Appliance-2 forwards this packet to server S1.
11. S1 processes the request packet and sends out a response packet. The destination IP address in the response packet is the IP address of client CL1, and the source IP address is the IP address of server S1.
12. The response packet reaches NS_Appliance-2.
13. NS_Appliance-2 encapsulates and encrypts the response packet in the same way that NS_Appliance-1 did with the request packet in steps 3-6.
14. NS_Appliance-2 sends the resulting packet to NS_Appliance-1.
15. NS_Appliance-1, upon receiving the packet from NS_Appliance-2, decrypts and decapsulates the packet in the same way that NS_Appliance-2 did with the request packet in steps 9-11.

Prerequisites for Configuring a CloudBridge Connector tunnel between two Datacenters

Before setting up a CloudBridge Connector tunnel, verify that the following tasks have been completed:

1. Deploy and set up a NetScaler appliance in each of the two datacenters.
2. Make sure that the CloudBridge Connector tunnel end-point IP addresses are accessible to each other.

Configuration Steps

To set up a CloudBridge Connector tunnel between a NetScaler appliance that resides in one datacenter and another NetScaler appliance that resides in the other datacenter, use the configuration utility or the command line interface of one of the NetScaler appliance.

When you use the configuration utility, the CloudBridge Connector tunnel configuration created on the first NetScaler appliance, is automatically pushed to the other endpoint (the other NetScaler appliance) of the CloudBridge Connector tunnel. Therefore, you do not have to access the configuration utility of the other NetScaler appliance to create the corresponding CloudBridge Connector tunnel configuration on it.

The CloudBridge Connector tunnel configuration on each of the NetScaler appliance consists of the following entities:

- **IPSec profile**. An IPSec profile entity specifies the IPSec protocol parameters, such as IKE version, encryption algorithm, hash algorithm, and PSK, to be used by the IPSec protocol in the CloudBridge Connector tunnel.
- **GRE tunnel**. An IP tunnel specifies the local IP address (a public SNIP address configured on the local NetScaler appliance), remote IP address (a public SNIP address configured on the remote NetScaler appliance), protocol (GRE) used to set up the CloudBridge Connector tunnel, and an IPSec profile entity.
- **Create a PBR rule and associate the IP tunnel with it**—A PBR entity specifies a set of conditions and an IP tunnel entity. The source IP address range and the destination IP range are the conditions for the PBR entity. You must set the source IP address range and the destination IP address range to specify the subnet whose traffic is to traverse the CloudBridge Connector tunnel. For example, consider a request packet that originates from a client on the subnet in the

first datacenter and is destined to a server on the subnet in the second datacenter. If this packet matches the source and destination IP address range of the PBR entity on the NetScaler appliance in the first datacenter, it is sent across the CloudBridge Connector tunnel associated with the PBR entity.

To create an IPSEC profile by using the command line interface

At the command prompt, type:

add ipsec profile <name> [-**ikeVersion** (V1 | V2)] [-**encAlgo** (AES | 3DES) ...] [-**hashAlgo** <hashAlgo> ...] [-**lifetime** <positive_integer>] (-**psk** | (-**publickey** <string> -**privatekey** <string> -**peerPublicKey**<string>)) [-**livenessCheckInterval** <positive_integer>] [-**replayWindowSize** <positive_integer>] [-**ikeRetryInterval** <positive_integer>] [-**retransmissiontime** <positive_integer>]

Example

add ipsec profile Cloud_Connector_DC1-DC2 -encAlgo AES -hashAlgo HMAC_SHA1

To create an IP tunnel and bind the IPSEC profile to it by using the command line interface

At the command prompt, type:

add ipTunnel <name> <remote> <remoteSubnetMask> <local> [-**protocol** <protocol>] [-**ipsecProfileName**<string>]

Example

add ipTunnel Cloud_Connector_DC1-DC2 203.0.113.133 255.255.255.0 198.51.100.15 –protocol GRE -ipsecProfileName Cloud_Connector_DC1-DC2

To create a PBR rule and bind the IPSEC tunnel to it by using the command line interface

At the command prompt, type:

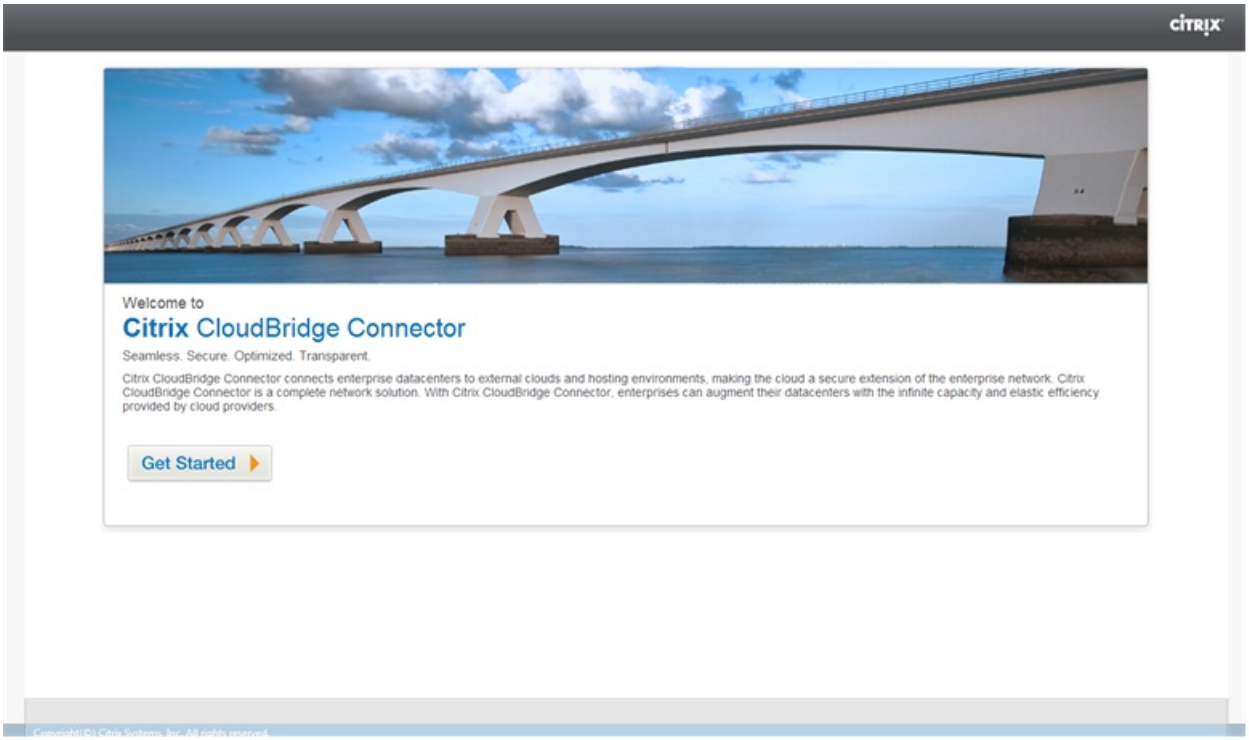
- **add ns pbr** <pbr_name> ALLOW -**srcIP** = <local_subnet_range> -**destIP** = <remote_subnet_range> -**ipTunnel** <tunnel_name>
- **apply ns pbrs**

Example

- add ns pbr PBR-DC1-DC2 ALLOW –srcIP 198.51.100.15 –destIP 203.0.113.133 ipTunnel Cloud_Connector_DC1-DC2
- apply ns pbrs

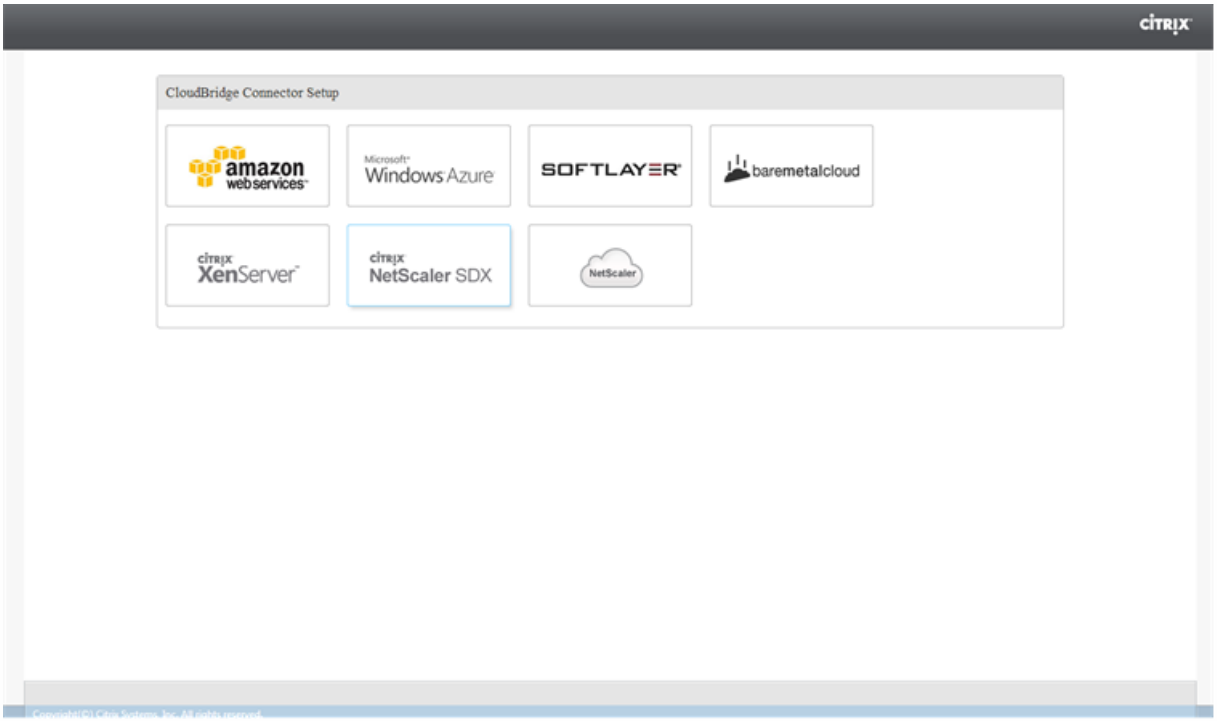
To configure a CloudBridge Connector tunnel in a NetScaler appliance by using the configuration utility

1. Type the NSIP address of a NetScaler appliance in the address line of a web browser.
2. Log on to the configuration utility of the NetScaler appliance by using your account credentials for the appliance.
3. Navigate to **System** > **CloudBridge Connector**.
4. In the right pane, under **Getting Started**, click **Create/Monitor CloudBridge**.
5. Click **Get Started.**



Note: If you already have any CloudBridge Connector tunnel configured on the NetScaler appliance, this screen does not appear, and you are taken to the **CloudBridge Connector Setup** pane.

6. In the **CloudBridge Connector Setup** pane, click **NetScaler**.



7. In the **NetScaler** pane, provide your account credentials for the remote NetScaler appliance. Click **Continue**.
8. In the **CloudBridge Connector Setting** pane, set the following parameter:
   - **CloudBridge Connector Name**— Name for the CloudBridge Connector configuration on the local appliance. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the

CloudBridge Connector configuration is created.

9. Under **Local Setting**, set the following parameter:
   - **Subnet IP**—IP address of the local endpoint of the CloudBridge Connector tunnel. Must be a public IP address of type SNIP.
10. Under **Remote Setting**, set the following parameter:
    - **Subnet IP**—IP address of the peer endpoint of the CloudBridge Connector tunnel. Must be a public IP address of type SNIP.
11. Under **PBR Setting**, set the following parameters:
    - **Operation**—Either the equals (=) or does not equal (!=) logical operator.
    - **Source IP Low***—Lower source IP address to match against the source IP address of an outgoing IPv4 packet.
    - **Source IP High**—Higher source IP address to match against the source IP address of an outgoing IPv4 packet.
    - **Operation**—Either the equals (=) or does not equal (!=) logical operator.
    - **Destination IP Low***—Lower destination IP address to match against the destination IP address of an outgoing IPv4 packet.
    - **Destination IP High**—Higher destination IP address to match against the destination IP address of an outgoing IPv4 packet.
12. (Optional) Under **Security Settings**, set the following IPSec protocol parameters to be used by the IPSec protocol in the CloudBridge Connector tunnel:
    - **Encryption Algorithm**—Encryption algorithm to be used by the IPSec protocol in the CloudBridge tunnel.
    - **Hash Algorithm—**Hash algorithm to be used by the IPSec protocol in the CloudBridge tunnel.
    - **Key**— Select one of the following IPSec authentication methods to be used by the two peers to mutually authenticate.
      - **Auto Generate Key**— Authentication based on a text string, called a pre-shared key (PSK), generated automatically by the local appliance. The PSKs keys of the peers are matched against each other for authentication.
      - **Specific Key—**Authentication based on a manually entered PSK. The PSKs of the peers are matched against each other for authentication.
        - **Pre Shared Security Key**—The text string entered for pre-shared key based authentication.
      - **Upload Certificates**—Authentication based on digital certificates.
        - **Public Key**—A local digital certificate to be used to authenticate the local NetScaler appliance to the peer before establishing IPSec security associations. The same certificate should be present and set for the Peer Public Key parameter in the peer.
        - **Private Key**—Private key of the local digital certificate.
        - **Peer Public Key**—Digital certificate of the peer. Used to authenticate the peer to the local end point before establishing IPSec security associations. The same certificate should be present and set for the Public key parameter in the peer.
13. Click **Done**.

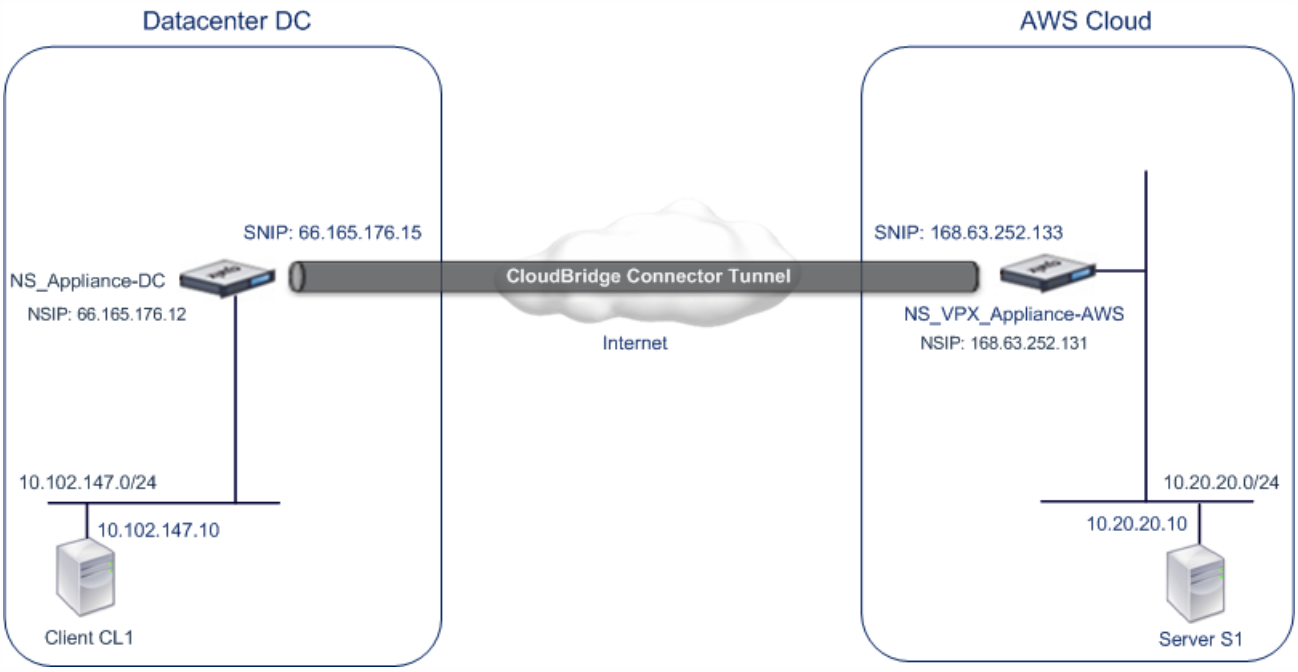
The new CloudBridge Connector tunnel configuration on both the NetScaler appliances appears on the Home tab of the respective configuration utility. The current status of the CloudBridge connector tunnel is indicated in the Configured CloudBridge Connectors pane. A green dot indicates that the tunnel is up. A red dot indicates that the tunnel is down.

Configuring CloudBridge Connector between Datacenter and AWS Cloud

Jan 28, 2011

You can configure a CloudBridge Connector tunnel between a datacenter and AWS cloud to leverage the infrastructure and computing capabilities of the data center and the AWS cloud. With AWS, you can extend your network without initial capital investment or the cost of maintaining the extended network infrastructure. You can scale your infrastructure up or down, as required. For example, you can lease more server capabilities when the demand increases.
To connect a datacenter to AWS cloud, you set up a CloudBridge Connector tunnel between a NetScaler appliance that resides in the datacenter and a NetScaler virtual appliance (VPX) that resides in AWS cloud.

As an illustration of a CloudBridge Connector tunnel between a datacenter and Amazon AWS cloud, consider an example in which a CloudBridge Connector tunnel is set up between NetScaler appliance NS_Appliance-DC, in datacenter DC, and NetScaler virtual appliance (VPX) NS_VPX_Appliance-AWS.



Both NS_Appliance-DC and NS_VPX_Appliance-AWS function in L3 mode. They enable communication between private networks in datacenter DC and the AWS cloud. NS_Appliance-DC and NS_VPX_Appliance-AWS enable communication between client CL1 in datacenter DC and server S1 in the AWS cloud through the CloudBridge Connector tunnel. Client CL1 and server S1 are on different private networks.

Note: AWS does not support L2 mode, hence it is necessary to have only L3 mode enabled on both the endpoints.
For proper communication between CL1 and S1, L3 mode is enabled on NS_Appliance-DC and NS_VPX_Appliance-AWS and routes are updated as such:

- CL1 have a route to NS_Appliance-DC for reaching S1
- NS_Appliance-DC have a route to NS_VPX_Appliance-AWS for reaching S1
- S1 should have a route to NS_VPX_Appliance-AWS for reaching CL1
- NS_VPX_Appliance-AWS have a route to NS_Appliance-DC for reaching CL1

The following table lists the settings on NetScaler appliance NS_Appliance-DC in datacenter DC.

Entity	Name	Details
The NSIP address		66.165.176.12
SNIP address		66.165.176.15
CloudBridge Connector tunnel	CC_Tunnel_DC-AWS	• Local endpoint IP address of the CloudBridge Connector tunnel =66.165.176.15 • Remote endpoint IP address of the CloudBridge Connector tunnel =168.63.252.133 **GRE Tunnel Details** • Name= CC_Tunnel_DC-AWS **IPSec Profile Details** • Name= CC_Tunnel_DC-AWS • Encryption algorithm= AES • Hash algorithm= HMAC SHA1

The following table lists the settings on NetScaler VPX NS_VPX_Appliance-AWS on AWS cloud.

Entity	Name	Details
NSIP address		10.102.25.30
Public EIP address mapped to the NSIP address		168.63.252.131
SNIP address		10.102.29.30
Public EIP address mapped to the SNIP address		168.63.252.133
CloudBridge Connector tunnel	CC_Tunnel_DC-AWS	• Local endpoint IP address of the CloudBridge Connector tunnel =168.63.252.133 • Remote endpoint IP address of the CloudBridge Connector tunnel = 66.165.176.15 **GRE Tunnel Details** • Name= CC_Tunnel_DC-AWS **IPSec Profile Details** • Name= CC_Tunnel_DC-AWS • Encryption algorithm= AES • Hash algorithm= HMAC SHA1

Prerequisites

Updated: 2015-06-01

Before setting up a CloudBridge Connector tunnel, verify that the following tasks have been completed:

1. Install, configure, and launch an instance of NetScaler Virtual appliance (VPX) on AWS cloud. For instructions on installing NetScaler VPX on AWS, see http://support.citrix.com/proddocs/topic/netscaler-vpx-10-5/nsvpx-aws-ns-vpxaws-con.html.

2. Deploy and configure a NetScaler physical appliance, or provisioning and configuring a NetScaler virtual appliance (VPX) on a virtualization platform in the datacenter.
   - For instructions on installing NetScaler vitrual appliances on Xenserver, see http://support.citrix.com/proddocs/topic/netscaler-vpx-10-5/ns-vpx-install-wrapper-con.html.
   - For instructions on installing NetScaler vitrual appliances on VMware ESX or ESXi, see http://support.citrix.com/proddocs/topic/netscaler-vpx-10-5/ns-vpx-install-on-esx-wrapper-con.html.
   - For instructions on installing NetScaler vitrual appliances on Microsoft Hyper-V, see http://support.citrix.com/proddocs/topic/netscaler-vpx-10-5/ns-vpx-install-on-msft-hyperv-wrapper-con.html.
3. Make sure that the CloudBridge Connector tunnel end-point IP addresses are accessible to each other.

NetScaler VPX License

After the initial instance launch, NetScaler VPX for AWS requires a license. If you are bringing your own license (BYOL), see the VPX Licensing Guide at: http://support.citrix.com/article/CTX122426.

You have to:
1. Use the licensing portal within MyCitrix to generate a valid license.
2. Upload the license to the instance.

If this is a **paid** marketplace instance, then you do not need to install a license. The correct feature set and performance will activate automatically.

Configuration Steps

To set up a CloudBridge Connector tunnel between a NetScaler appliance that resides in a datacenter and a NetScaler virtual appliance (VPX) that resides on the AWS cloud, use the configuration utility of the NetScaler appliance.

When you use the configuration utility, the CloudBridge Connector tunnel configuration created on the NetScaler appliance, is automatically pushed to the other endpoint or peer (the NetScaler VPX on AWS) of the CloudBridge Connector tunnel. Therefore, you do not have to access the configuration utility (GUI) of the NetScaler VPX on AWS to create the corresponding CloudBridge Connector tunnel configuration on it.

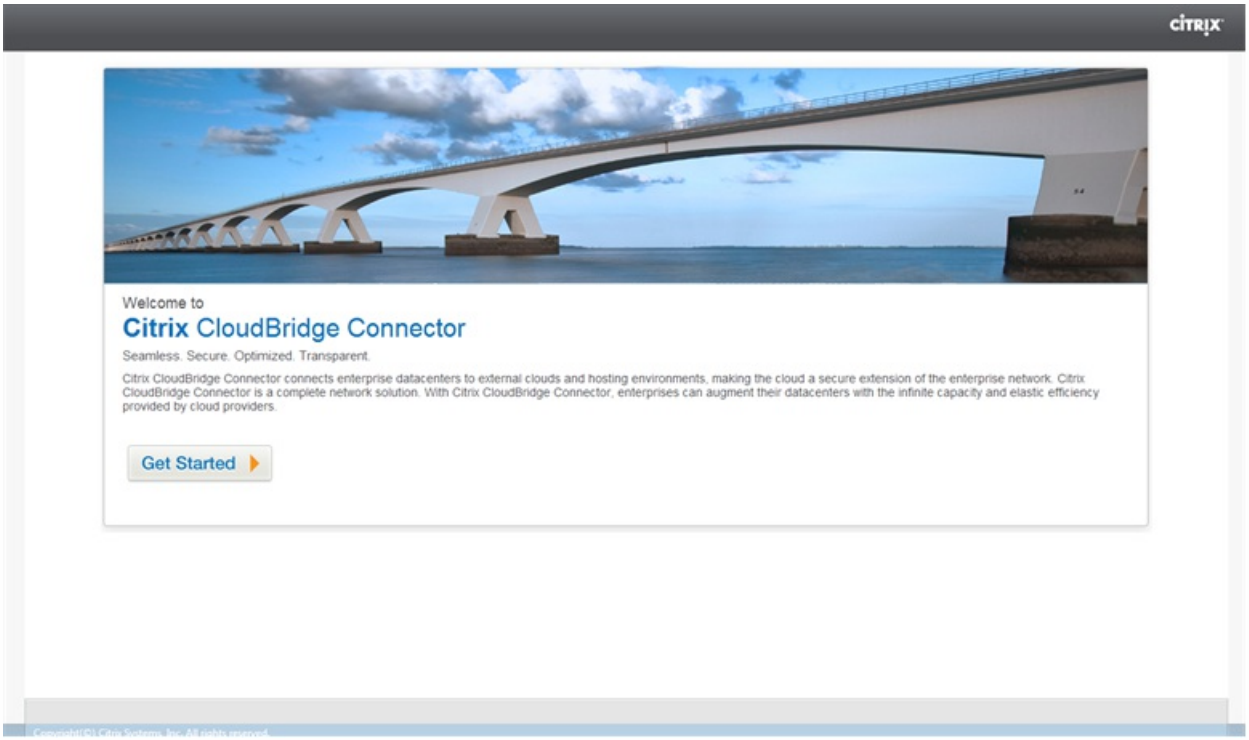
The CloudBridge Connector tunnel configuration on both peers (the NetScaler appliance that resides in the datacenter and the NetScaler virtual appliance (VPX) that resides on the AWS cloud) consists of the following entities:
- **IPSec profile**. An IPSec profile entity specifies the IPSec protocol parameters, such as IKE version, encryption algorithm, hash algorithm, and PSK, to be used by the IPSec protocol in both the peers of the CloudBridge Connector tunnel.
- **GRE tunnel**. An IP tunnel specifies a local IP address (a public SNIP address configured on the local peer), remote IP address (a public SNIP address configured on the remote peer), protocol (GRE) used to set up the CloudBridge Connector tunnel, and an IPSec profile entity.

- **Netbridge**. A logical container that holds or represents the CloudBridge Connector tunnel configuration on each of the peers. A GRE tunnel entity is associated with the netbridge. A particular CloudBridge Connector tunnel configuration on a peer is identified by the name of the netbridge entity.

To configure a CloudBridge Connector tunnel in a NetScaler appliance by using the configuration utility

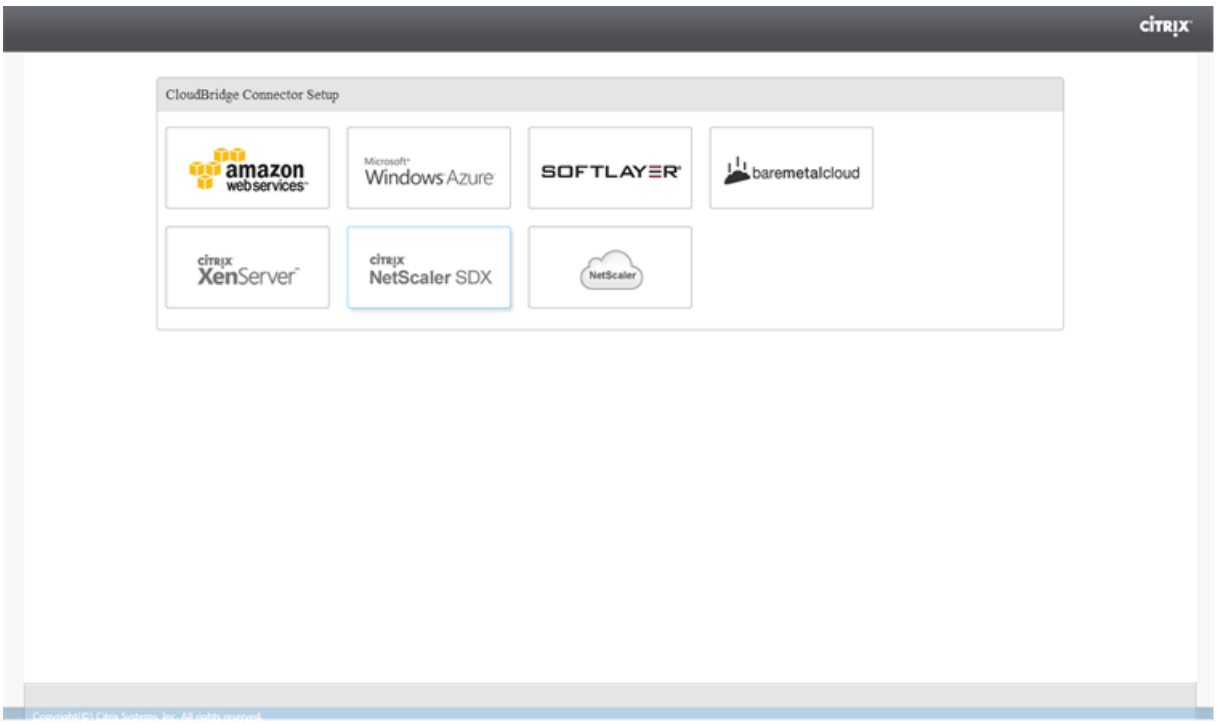
1. Type the NSIP address of a NetScaler appliance in the address line of a web browser.
2. Log on to the configuration utility of the NetScaler appliance by using your account credentials for the appliance.

3. Navigate to System > CloudBridge Connector.
4. In the right pane, under **Getting Started**, click **Create/Monitor CloudBridge**.
5. Click **Get Started**.



Note: If you already have any CloudBridge Connector tunnel configured on the NetScaler appliance, this screen does not appear, and you are taken to the **CloudBridge Connector Setup** pane.

6. In the **CloudBridge Connector Setup** pane, click **amazon web services**.



7. In the **Amazon** pane, provide your AWS account credentials: AWS Access Key ID and AWS Secret Access Key. You can obtain these access keys from the AWS GUI console. Click **Continue**.
8. In the **NetScaler** pane, select the NSIP address of the NetScaler virtual appliance running on AWS. Then, provide your account credentials for the NetScaler virtual appliance. Click **Continue**.
9. In the **CloudBridge Connector Setting** pane, set the following parameter:
   - **CloudBridge Connector Name**— Name for the CloudBridge Connector configuration on the local appliance. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the CloudBridge Connector configuration is created.
10. Under **Local Setting**, set the following parameter:
    - **Subnet IP**—IP address of the local endpoint of the CloudBridge Connector tunnel. Must be a public IP address of type SNIP.
11. Under **Remote Setting**, set the following parameter:
    - **Subnet IP**— IP address of the CloudBridge Connector tunnel end point on the AWS side. Must be an IP address of type SNIP on the NetScaler VPX instance on AWS.
    - **NAT**—Public IP address (EIP) in AWS that is mapped to the SNIP configured on the NetScaler VPX instance on AWS.
12. (Optional) Under **Security Settings**, set the following IPSec protocol parameters to be used by the IPSec protocol in the CloudBridge Connector tunnel:
    - **Encryption Algorithm**— Encryption algorithm to be used by the IPSec protocol in the CloudBridge tunnel.
    - **Hash Algorithm—**Hash algorithm to be used by the IPSec protocol in the CloudBridge tunnel.
    - **Key**— Select one of the following IPSec authentication methods to be used by the two peers to mutually authenticate.
      - **Auto Generate Key**— Authentication based on a text string, called a pre-shared key (PSK), generated automatically by the local appliance. The PSKs keys of the peers are matched against each other for

authentication.

- **Specific Key—**Authentication based on a manually entered PSK. The PSKs of the peers are matched against each other for authentication.
  - **Pre Shared Security Key**—The text string entered for pre-shared key based authentication.
- **Upload Certificates**—Authentication based on digital certificates.
  - **Public Key**—A local digital certificate to be used to authenticate the local peer to the remote peer before establishing IPSec security associations. The same certificate should be present and set for the Peer Public Key parameter in the peer.
  - **Private Key**—Private key of the local digital certificate.
  - **Peer Public Key**—Digital certificate of the peer. Used to authenticate the peer to the local end point before establishing IPSec security associations. The same certificate should be present and set for the Public key parameter in the peer.

13. Click **Done**.

The new CloudBridge Connector tunnel configuration on the NetScaler appliance in the datacenter appears on the Home tab of the configuration utility.

The corresponding new CloudBridge Connector tunnel configuration on the NetScaler VPX appliance in the AWS cloud appears on the configuration utility.

The current status of the CloudBridge connector tunnel is indicated in the Configured CloudBridge pane. A green dot indicates that the tunnel is up. A red dot indicates that the tunnel is down.

Configuring a CloudBridge Connector Tunnel Between a Datacenter and Azure Cloud

Sep 30, 2015

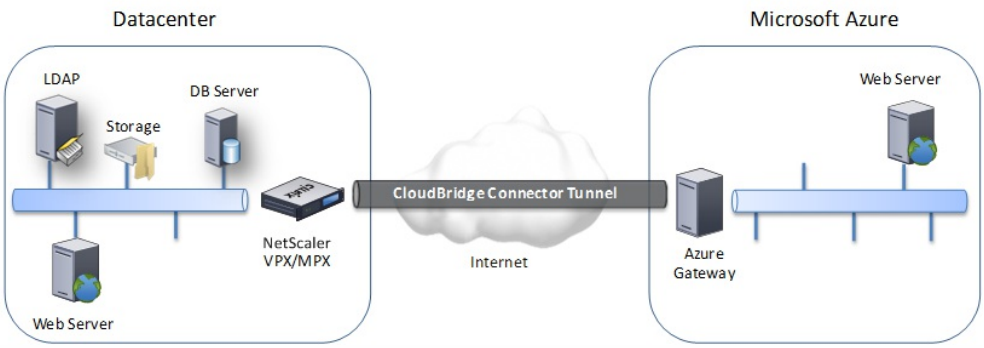
The NetScaler appliance provides connectivity between your enterprise datacenters and the Microsoft cloud hosting provider, Azure, making Azure a seamless extension of the enterprise network. NetScaler encrypts the connection between the enterprise datacenter and Azure cloud so that all data transferred between the two is secure.

This section includes the following:

- How CloudBridge Connector Tunnel Works
- Example of CloudBridge Connector Tunnel Configuration and Data Flow
- Points to Consider for a CloudBridge Connector tunnel Configuration
- Configuring the CloudBridge Connector Tunnel
- Monitoring the CloudBridge Connector Tunnel

How CloudBridge Connector Tunnel Works

To connect a datacenter to AWS cloud, you set up a CloudBridge Connector tunnel between a NetScaler appliance that resides in the datacenter and a gateway that resides in the Azure cloud. The NetScaler appliance in the datacenter and the gateway in Azure cloud are the end points of the CloudBridge Connector tunnel and are called peers of the CloudBridge Connector tunnel.



A CloudBridge Connector tunnel between a datacenter and Azure cloud uses the open-standard Internet Protocol security (IPSec) protocol suite, in tunnel mode, to secure communications between peers in the CloudBridge Connector tunnel. In a CloudBridge Connector tunnel, IPSec ensures:

- Data integrity
- Data origin authentication
- Data confidentiality (encryption)
- Protection against replay attacks

IPSec uses the tunnel mode in which the complete IP packet is encrypted and then encapsulated. The encryption uses the Encapsulating Security Payload (ESP) protocol, which ensures the integrity of the packet by using a HMAC hash function and ensures confidentiality by using an encryption algorithm. The ESP protocol, after encrypting the payload and calculating the HMAC, generates an ESP header and inserts it before the encrypted IP packet. The ESP protocol also generates an ESP trailer and inserts it at the end of the packet.

The IPSec protocol then encapsulates the resulting packet by adding an IP header before the ESP header. In the IP header, the destination IP address is set to the IP address of the CloudBridge Connecter peer.

Peers in the CloudBridge Connector tunnel use the Internet Key Exchange version 1 (IKEv1) protocol (part of the IPSec protocol suite) to negotiate secure communication, as follows:

1. The two peers mutually authenticate with each other, using pre-shared key authentication, in which the peers exchange a text string called a pre-shared key (PSK). The pre-shared keys are matched against each other for authentication. Therefore, for the authentication to be successful, you must configure the same pre-shared key on each of the peers.
2. The peers then negotiate to reach agreement on:
   - An encryption algorithm
   - Cryptographic keys for encrypting data on one peer and decrypting it on the other.

   This agreement upon the security protocol, encryption algorithm and cryptographic keys is called a Security Association (SA). SAs are one-way (simplex). For example, when a CloudBridge Connector tunnel is set up between a NetScaler appliance in a datacenter and a gateway in an Azure cloud, both the datacenter appliance and the Azure gateway have two SAs. One SA is used for processing out-bound packets, and the other SA is used for processing inbound packets. SAs expire after a specified interval of time, which is called the lifetime.

Example of CloudBridge Connector Tunnel Configuration and Data Flow

As an illustration of CloudBridge Connector Tunnel, consider an example in which a CloudBridge Connector tunnel is set up between NetScaler appliance CB_Appliance-1 in a datacenter and gateway Azure_Gateway-1 in Azure cloud.

CB_Appliance-1 also functions as an L3 router, which enables a private network in the datacenter to reach a private network in the Azure cloud through the CloudBridge Connector tunnel. As a router, CB_Appliance-1 enables communication between client CL1 in the datacenter and server S1 in the Azure cloud through the CloudBridge Connector tunnel. Client CL1 and server S1 are on different private networks.

On CB_Appliance-1, the CloudBridge Connector tunnel configuration includes an IPSec profile entity named CB_Azure_IPSec_Profile, a CloudBridge Connector tunnel entity named CB_Azure_Tunnel, and a policy based routing (PBR) entity named CB_Azure_Pbr.

The IPSec profile entity CB_Azure_IPSec_Profile specifies the IPSec protocol parameters, such as IKE version, encryption algorithm, and hash algorithm, to be used by the IPSec protocol in the CloudBridge Connector tunnel. CB_Azure_IPSec_Profile is bound to IP tunnel entity CB_Azure_Tunnel.

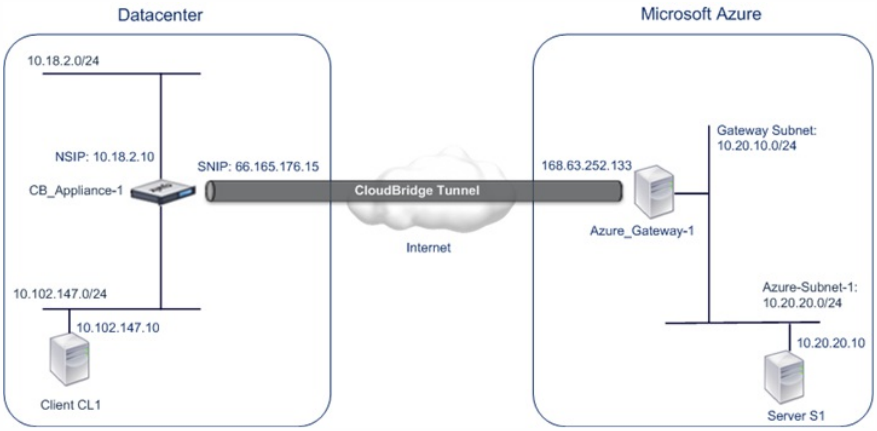
CloudBridge Connector tunnel entity CB_Azure_Tunnel specifies the local IP address (a public IP (SNIP) address configured on the NetScaler appliance), the remote IP address (the IP address of the Azure_Gateway-1), and the protocol (IPSec) used to set up the CloudBridge Connector tunnel. CB_Azure_Tunnel is bound to the PBR entity CB_Azure_Pbr.

The PBR entity CB_Azure_Pbr specifies a set of conditions and a CloudBridge Connector tunnel entity (CB_Azure_Tunnel). The source IP address range and the destination IP address range are the conditions for CB_Azure_Pbr. The source IP address range and the destination IP address range are specified as a subnet in the datacenter and a subnet in the Azure cloud, respectively. Any request packet originating from a client in the subnet in the datacenter and destined to a server in the subnet on the Azure cloud matches the conditions in CB_Azure_Pbr. This packet is then considered for CloudBridge processing and is sent across the CloudBridge Connector tunnel (CB_Azure_Tunnel) bound to the PBR entity.

On Microsoft Azure, the CloudBridge Connector tunnel configuration includes a local network entity named My-Datacenter-Network, a virtual network entity named Azure-Network-for-CloudBridge-Tunnel, and a gateway named Azure_Gateway-1.

The local (local to Azure) network entity My-Datacenter-Network specifies the IP address of the NetScaler appliance on the datacenter side, and the datacenter subnet whose traffic is to traverse the CloudBridge Connector tunnel. The virtual network entity Azure-Network-for-CloudBridge-Tunnel defines a private subnet named Azure-Subnet-1 in Azure. The traffic of the subnet traverses the CloudBridge Connector tunnel. The server S1 is provisioned in this subnet.

The local network entity My-Datacenter-Network is associated with the virtual network entity Azure-Network-for-CloudBridge-Tunnel. This association defines the remote and local network details of the CloudBridge Connector tunnel configuration in Azure. Gateway Azure_Gateway-1 was created for this association to become the CloudBridge end point at the Azure end of the CloudBridge Connector tunnel.



The following table lists the settings used in this example.

Entity	Name	Details
Settings highlight of the CloudBridge Connector tunnel setup		
IP address of the CloudBridge Connector tunnel end point (CB_Appliance-1) in the datacenter side	66.165.176.15	
IP address of the CloudBridge Connector tunnel end point (Azure_Gateway-1) in the Azure	168.63.252.133	
Datacenter Subnet, the traffic of which is to traverse the CloudBridge Connector tunnel	10.102.147.0/24	
Azure Subnet, the traffic of which is to traverse the CloudBridge Connector tunnel	10.20.0.0/16	

Entity	Name	Details
Settings on NetScaler appliance CB_Appliance-1 in Datacenter		
	SNIP1(for reference purposes only)	66.165.176.15
IPSec profile	CB_Azure_IPSec_Profile	• IKE version = v1 • Encryption algorithm = AES • Hash algorithm = HMAC SHA1
CloudBridge Connector tunnel	CB_Azure_Tunnel	• Remote IP = 168.63.252.133 • Local IP= 66.165.176.15 • Tunnel protocol = IPSec • IPSec profile= CB_Azure_IPSec_Profile
Policy based route	CB_Azure_Pbr	• Source IP range = Subnet in the datacenter =10.102.147.0-10.102.147.255 • Destination IP range =Subnet in Azure =10.20.0.0-10.20.255.255 • IP Tunnel = CB_Azure_Tunnel
Settings on Microsoft Azure		
Public IP Address of the Azure_Gateway-1		168.63.252.133
Local Network	My-Datacenter-Network	• VPN Device IP address =SNIP address of the NetScaler appliance = 66.165.176.15 • Address space= Subnet in datacenter =10.102.147.0/24
Virtual Network	Azure-Network-for-CloudBridge-Tunnel	• Address Space= 10.20.0.0/16 • Subnet in Azure=Azure-Subnet-1= 10.20.20.0/24 • Local Network=My-Datacenter-Network • Gateway Subnet=10.20.10.0/24

Following is the traffic flow in the CloudBridge Connector tunnel:

1. Client C1 sends a request to server S1.
2. The request reaches NetScaler appliance CB_Appliance-1.
3. The request packet in CB_Appliance-1 matches the condition specified in the PBR entity CB_Azure_Pbr as the source IP address and the destination IP address of the request packet belonging to the source IP range and destination IP range, respectively, set in CB_Azure_Pbr.
4. Because CloudBridge Connector tunnel entity CB_Azure_Tunnel is bound to CB_Azure_Pbr, the appliance prepares the packet to be sent across the CB_Azure_Tunnel.
5. For CloudBridge Connector tunnel CB_Azure_Tunnel, CB_Appliance-1 checks the stored IPSec security association (SA) parameters for processing outbound packets, as agreed between CB_Appliance-1 in the datacenter and Azure_Gateway-1 in the Azure cloud. The IPSec Encapsulating Security Payload (ESP) protocol in the NetScaler appliance uses these SA parameters for outbound packets to encrypt the request packet.
6. The ESP protocol ensures the packet's integrity by using a HMAC hash function and the packet's confidentiality by using the AES encryption algorithm. The ESP protocol, after encrypting the request packet and calculating the HMAC, generates an ESP header and then inserts it before the encrypted IP packet. The ESP protocol also generates an ESP trailer and then inserts it at the end of the encrypted IP packet.
7. The IPSec protocol encapsulates the resulting packet by adding an IP header before the ESP header. The destination address in the IP header is the IP address of Azure-gateway-1, and the source address is the SNIP2 address.
8. The resulting packet is sent to Azure_Gateway-1. There is
9. Azure-gateway-1, upon receiving the packet from CB_Appliance-1, decapsulates the packet by removing the IPSec IP header.
10. Azure-gateway-1 then checks the stored IPSec security association (SA) parameters for processing inbound packets, as agreed between CB_Appliance-1 and Azure_Gateway-1. The IPSec ESP protocol on Azure_Gateway-1 uses these SA parameters for inbound packets, and the ESP header of the decapsulated request packet, to decrypt the packet.

11. The resulting packet is the same packet as the one received by CB_Appliance-1 in step 2. This packet has the destination IP address set to the IP address of server S1. Azure_Gateway-1 forwards this packet to server S1.
12. S1 processes the request packet and sends out a response packet. The destination IP address in the response packet is the IP address of client CL1, and source IP address is the IP address of server S1.
13. The response packet reaches Azure_Gateway-1. Microsoft Azure checks the stored IPSec security association (SA) parameters for processing outbound packets, as agreed between CB_Appliance-1 and Azure_Gateway-1. Microsoft Azure encrypts and encapsulates the response packet in the same way that CB_Appliance-1 encrypted and encapsulated the request packet in steps 5, 6, and 7.
14. Azure_Gateway-1 sends the resulting packet to CB_Appliance-1.
15. CB_Appliance-1, upon receiving the packet from Azure_Gateway-1, decapsulates and decrypts the packet in the same way that Azure_Gateway-1 decapsulated and decrypted the request packet in steps 9 and 10.
16. The resulting packet is the same packet that was received by Azure_Gateway-1 in step 13. This response packet has the destination IP address set to the IP address of server CL1. CB_Appliance-1 forwards the response packet to client CL1.

Points to Consider for a CloudBridge Connector tunnel Configuration

Updated: 2014-04-15

Before configuring a CloudBridge Connector tunnel between a NetScaler appliance in datacenter and Microsoft Azure, consider the following points:

1. The NetScaler appliance must have a public facing IPv4 address (type SNIP) to use as a tunnel end-point address for the CloudBridge Connector tunnel. Also, the NetScaler appliance should not be behind a NAT device.
2. Azure supports the following IPSec settings for a CloudBridge Connector tunnel. Therefore, you must specify the same IPSec settings while configuring the NetScaler for the CloudBridge Connector tunnel.
   - IKE version = v1
   - Encryption algorithm = AES
   - Hash algorithm = HMAC SHA1
3. You must configure the firewall in the datacenter edge to allow the following.
   - Any UDP packets for port 500
   - Any UDP packets for port 4500
   - Any ESP (IP protocol number 50) packets
4. IKE re-keying, which is renegotiation of new cryptographic keys between the CloudBridge Connector tunnel end points to establish new SAs, is not supported. When the Security Associations (SAs) expire, the tunnel goes into the DOWN state. Therefore, you must set a very large value for the lifetimes of SAs.
5. You must configure Microsoft Azure before specifying the tunnel configuration on the NetScaler, because the public IP address of the Azure end (gateway) of the tunnel, and the PSK, are automatically generated when you set up the tunnel configuration in Azure. You need this information for specifying the tunnel configuration on the NetScaler.

Configuring the CloudBridge Connector Tunnel

Updated: 2014-04-15

For setting up a CloudBridge Connector tunnel between your datacenter and Azure, you must install CloudBridge VPX/MPX in your datacenter, configure Microsoft Azure for the CloudBridge Connector tunnel, and then configure the NetScaler appliance in the data center for the CloudBridge Connector tunnel.

Configuring a CloudBridge Connector tunnel between a NetScaler appliance in datacenter and Microsoft Azure consists of the following tasks:

1. **Setting up the NetScaler appliance in the datacenter**. This task involves deploying and configuring a NetScaler physical appliance (MPX), or provisioning and configuring a NetScaler virtual appliance (VPX) on a virtualization platform in the datacenter.
2. **Configuring Microsoft Azure for the CloudBridge Connector tunnel**. This task involves creating local network, virtual network, and gateway entities in Azure. The local network entity specifies the IP address of the CloudBridge Connector tunnel end point (the NetScaler appliance) on the datacenter side, and the datacenter subnet whose traffic is to traverse the CloudBridge Connector tunnel. The virtual network defines a network on Azure. Creating the virtual network includes defining a subnet whose traffic is to traverse the CloudBridge Connector tunnel to be formed. You then associate the local network with the virtual network. Finally, you create a gateway that becomes the end point at the Azure end of the CloudBridge Connector tunnel.
3. **Configuring the NetScaler appliance in the datacenter for the CloudBridge Connector tunnel**. This task involves creating an IPSec profile, an IP tunnel entity, and a PBR entity in the NetScaler appliance in datacenter. The IPSec profile entity specifies the IPSec protocol parameters, such as IKE version, encryption algorithm, hash algorithm, and PSK, to be used in the CloudBridge Connector tunnel. The IP tunnel specifies the IP address of both the CloudBridge Connector tunnel end points (the NetScaler appliance in datacenter and the gateway in Azure) and the protocol to be used in the CloudBridge Connector tunnel. You then associate the IPSec profile entity with the IP tunnel entity. The PBR entity specifies the two subnets, in the datacenter and in the Azure cloud, that are to communicate with each other through the CloudBridge Connector tunnel. You then associate the IP tunnel entity with the PBR entity.

Configuring Microsoft Azure for the CloudBridge Connector tunnel

Updated: 2014-04-15

To create a CloudBridge Connector tunnel configuration on Microsoft Azure, use the Microsoft Windows Azure Management Portal, which is a web based graphical interface for creating and managing resources on Microsoft Azure.

Before you begin the CloudBridge Connector tunnel configuration on Azure cloud, make sure that:

- You have a user account for Microsoft Azure.
- You have a conceptual understanding of Microsoft Azure.
- You are familiar with the Microsoft Windows Azure Management Portal.

Note: The procedures for configuring Microsoft Azure for a CloudBridge Connector tunnel might change over time, depending on the Microsoft Azure release cycle. Citrix recommends the following Microsoft Azure documentation for the latest procedures.

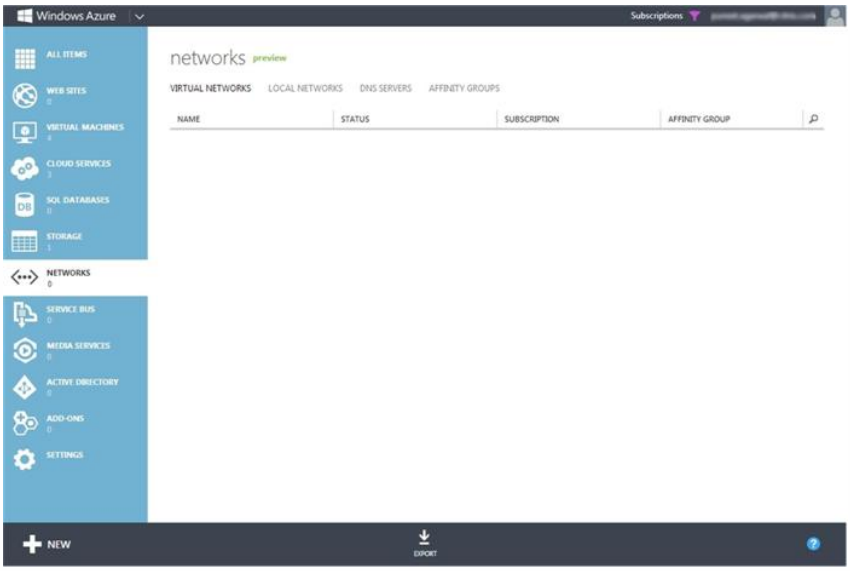
- http://www.windowsazure.com/en-us/manage/services/networking/cross-premises-connectivity/

To configure a CloudBridge Connector tunnel between a datacenter and an Azure cloud, perform the following tasks on Microsoft Azure by using the Microsoft Windows Azure Management Portal:

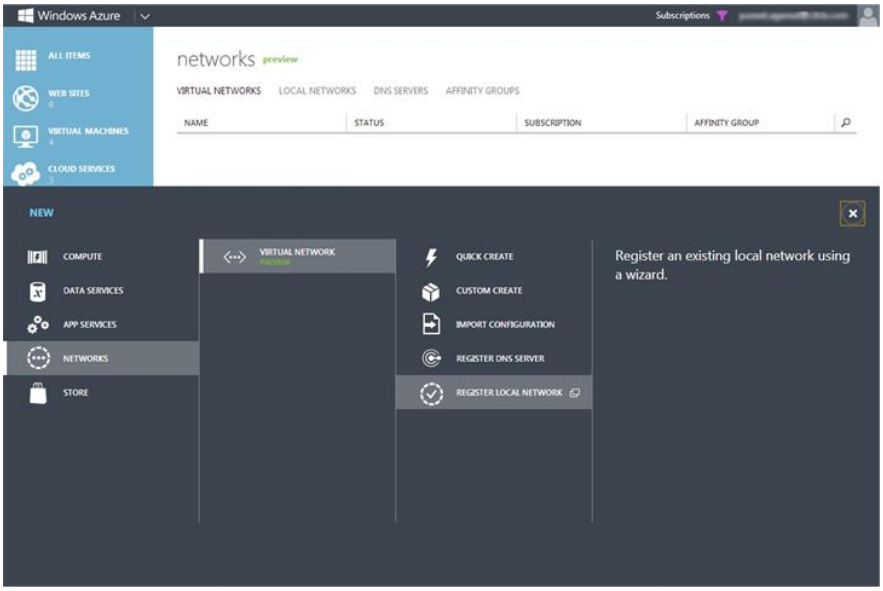
- **Create a local network entity**. Create a local network entity in Windows Azure for specifying the network details of the datacenter. A local network entity specifies the IP address of the CloudBridge Connector tunnel end point (the NetScaler) on the datacenter side and the datacenter subnet whose traffic is to traverse the CloudBridge Connector tunnel.
- **Create a Virtual Network**. Create virtual network entity that defines a network on Azure. This task includes defining a private address space, where you provide a range of private addresses and subnets belonging to the range specified in the address space. The traffic of the subnets will traverse the CloudBridge Connector tunnel. You then associate a local network entity with the virtual network entity. This association lets Azure create a configuration for a CloudBridge Connector tunnel between the virtual network and the data center network. A gateway (to be created) in Azure for this virtual network will be the CloudBridge end point at the Azure end of the CloudBridge Connector tunnel. You then define a private subnet for the gateway to be created. This subnet belongs to the range specified in the address space in the virtual network entity.
- **Create a gateway in Windows Azure**. Create a gateway that becomes the end point at the Azure end of the CloudBridge Connector tunnel. Azure, from its pool of public IP addresses, assigns an IP address to the gateway created.
- **Gather the public IP address of the gateway and the pre-shared key.** For a CloudBridge Connector tunnel configuration on Azure, the public IP address of the gateway and the pre-shared Key (PSK) are automatically generated by Azure. Make a note of this information. You will need it for configuring the CloudBridge Connector tunnel on the NetScaler in datacenter.

To specify a local network by using the Microsoft Windows Azure Management Portal

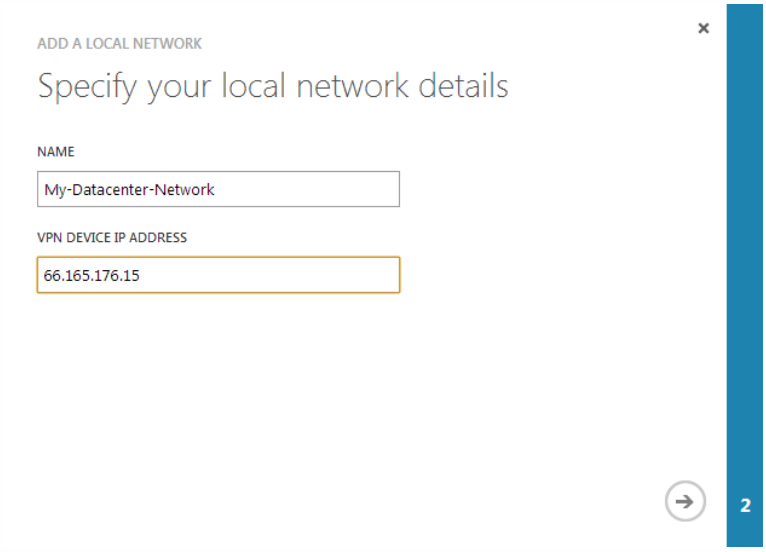
1. In the left pane, click NETWORKS.
2. In the lower left-hand corner of the screen, click + NEW.



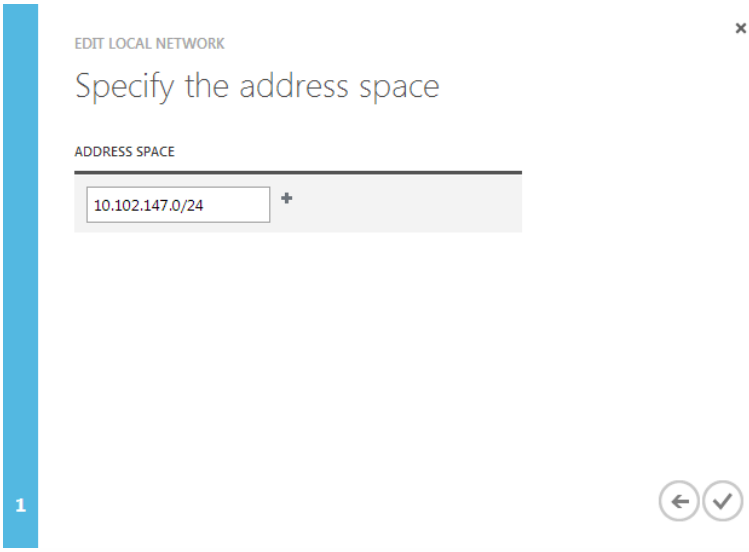
3. In the NEW navigation pane, click NETWORK, then click VIRTUAL NETWORK, and then click REGISTER LOCAL NETWORK.



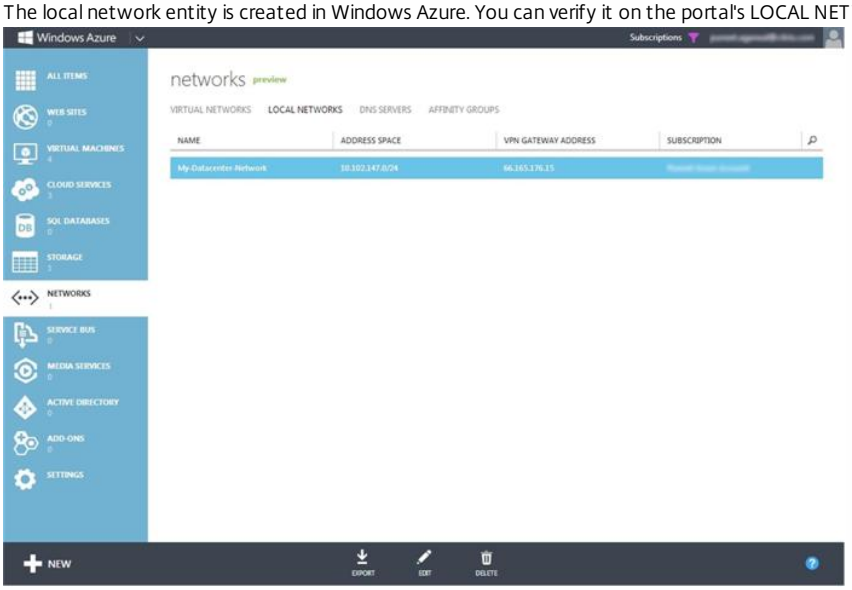
4. In the ADD A LOCAL NETWORK wizard, in the specify your local network details screen, set the following parameters:
   - NAME
   - VPN DEVICE IP ADDRESS



5. In the lower right corner of the screen, click -> (forward arrow mark).
6. On the Specify the address space screen, set the following parameter:
   - ADDRESS SPACE

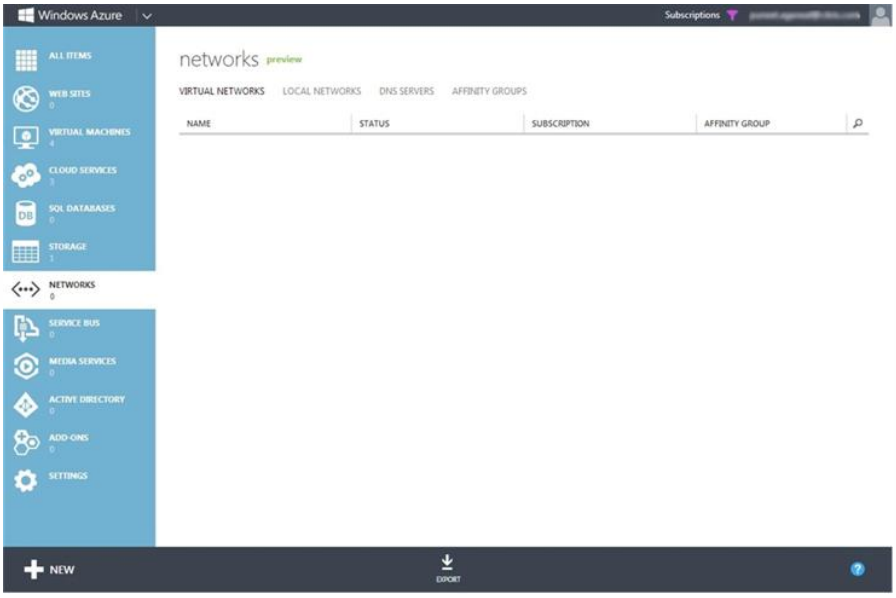


7. In the lower right corner of the screen, click the check mark.
8. The local network entity is created in Windows Azure. You can verify it on the portal's LOCAL NETWORK tab.

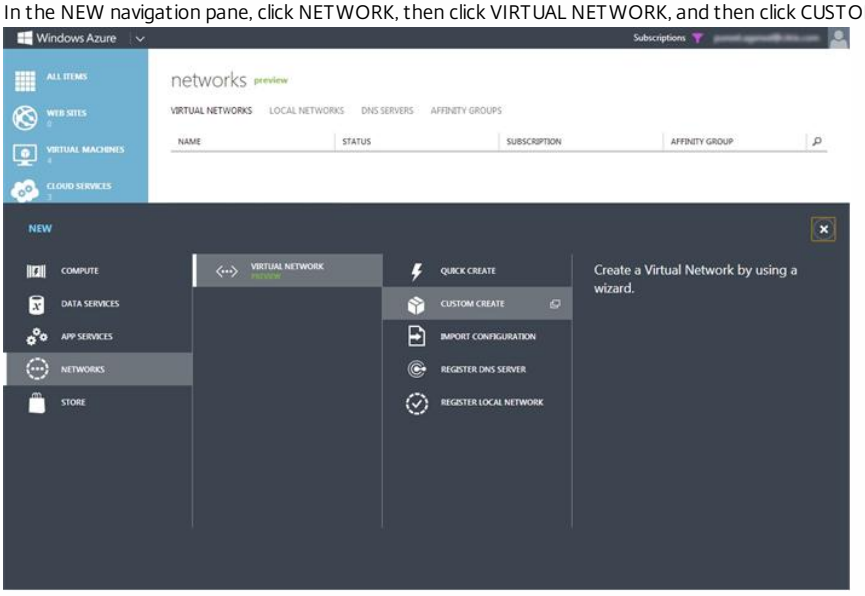


To create a virtual network in Azure by using the Microsoft Windows Azure Management Portal

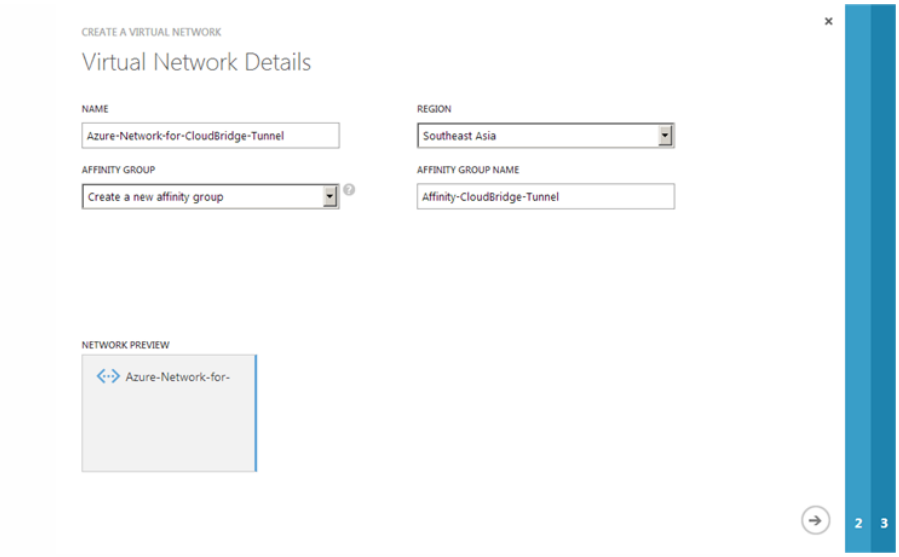
1. In the left pane, click NETWORKS.
2. In the lower left-hand corner of the screen, click + New.



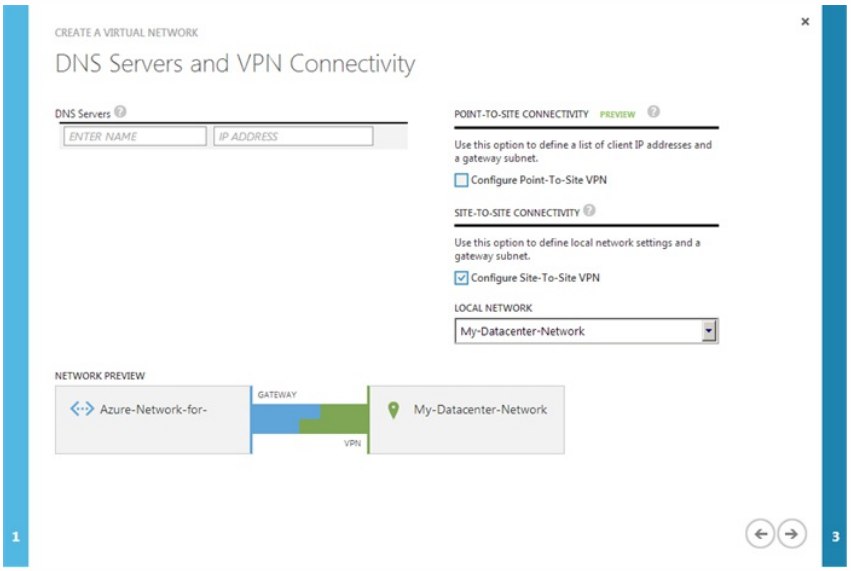
3. In the NEW navigation pane, click NETWORK, then click VIRTUAL NETWORK, and then click CUSTOM CREATE.



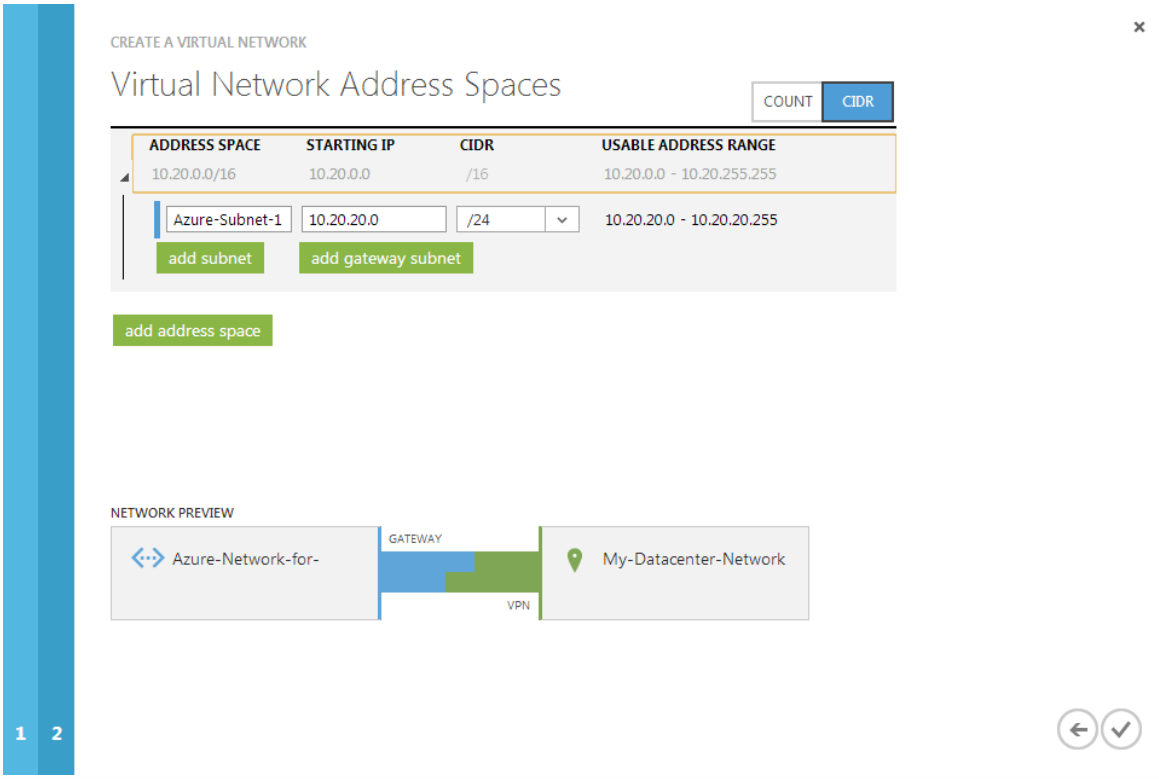
4. In the CREATE A VIRTUAL NETWORK wizard, in the Virtual Network Details screen, set the following parameters:
   - NAME
   - AFFINITY GROUP
   - REGION
   - AFFINITY GROUP NAME

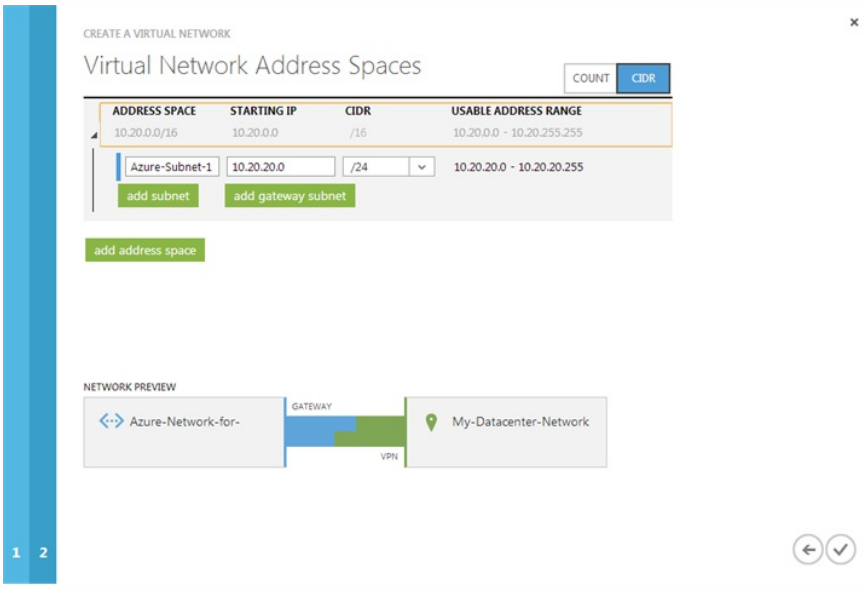


5. Click -> (forward arrow mark) in the lower right-hand corner of the screen.
6. In the DNS Servers and VPN Connectivity screen, in SITE-TO-SITE CONNECTIVITY, select Configure Site-To-Site VPN and set the following parameter:
   - LOCAL NETWORK

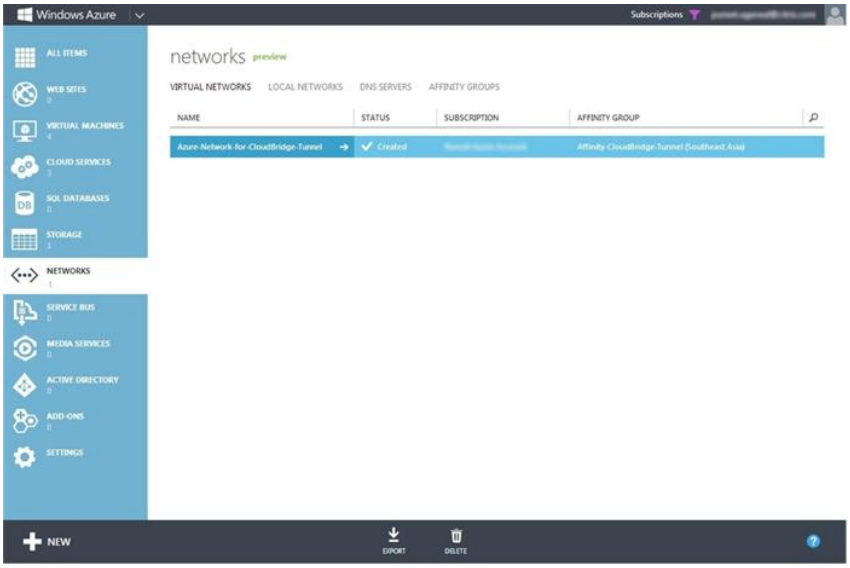


7. In the Address Space and Subnets screen, set the following parameters:
   - ADDRESS SPACE
   - SUBNETS
   - Gateway



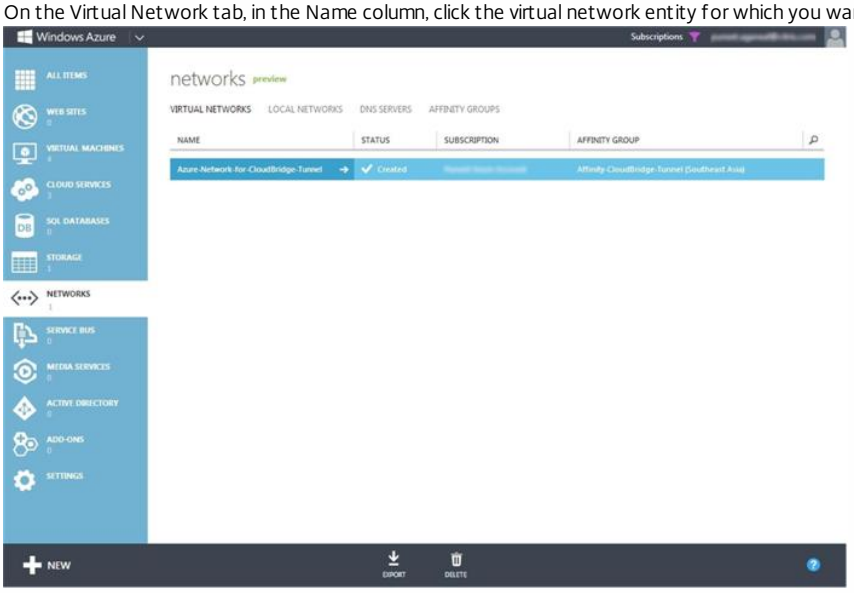


8. Click the check mark in the lower right-hand corner of the screen.
9. The virtual network is created in Windows Azure and is listed on the VIRTUAL NETWORK tab.

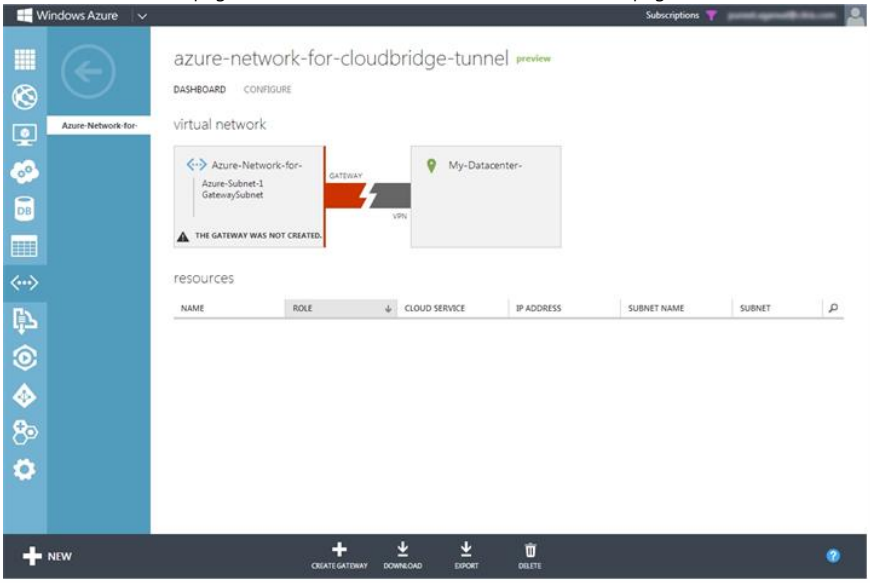


To create a gateway by using the Microsoft Windows Azure Management Portal

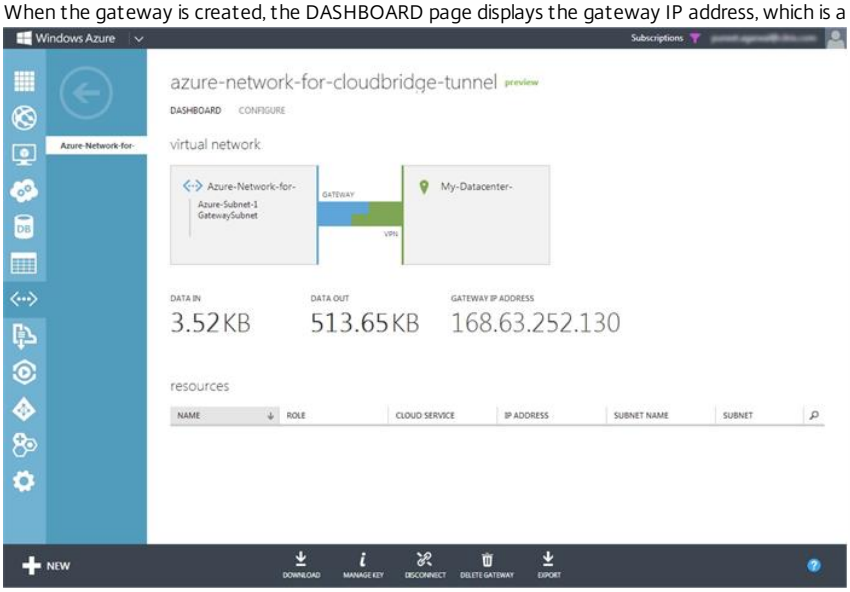
1. In the left pane, click NETWORKS.
2. On the Virtual Network tab, in the Name column, click the virtual network entity for which you want to create a gateway.



3. On the DASHBOARD page of the virtual network, at the bottom of the page, click + Create Gateway.

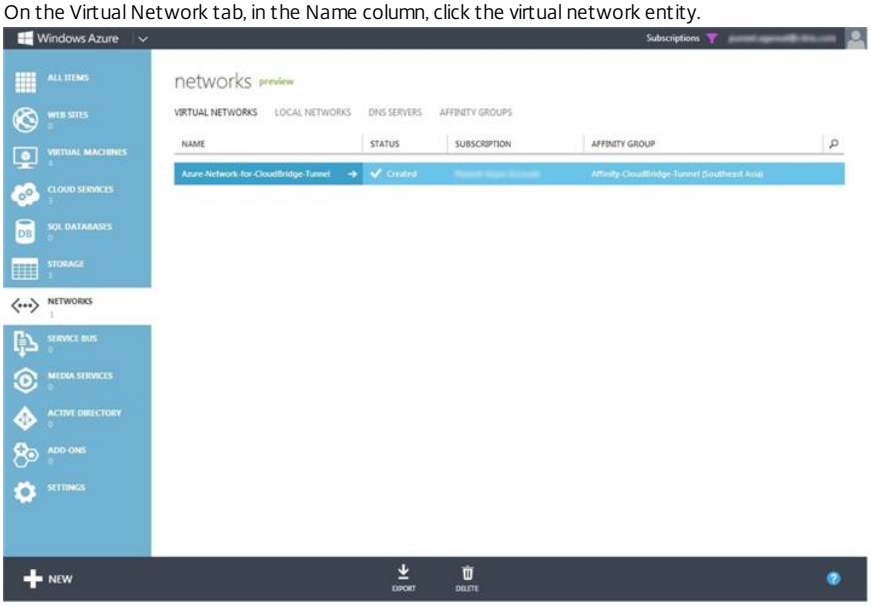


4. When prompted to confirm you want the gateway created, click YES. Creating the gateway can take up to 15 minutes.
5. When the gateway is created, the DASHBOARD page displays the gateway IP address, which is a public IP address.

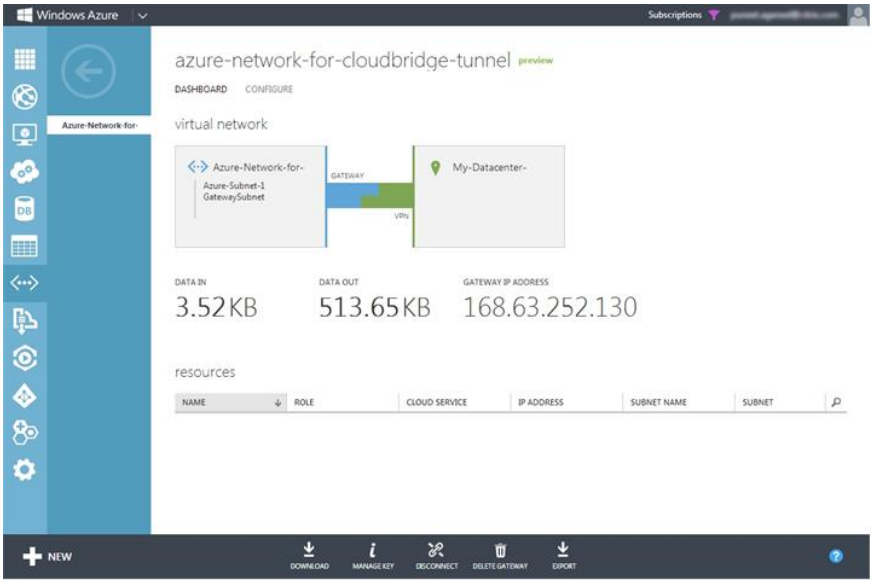


To gather public IP address of the gateway and the pre-shared key information by using the Microsoft Windows Azure Management Portal

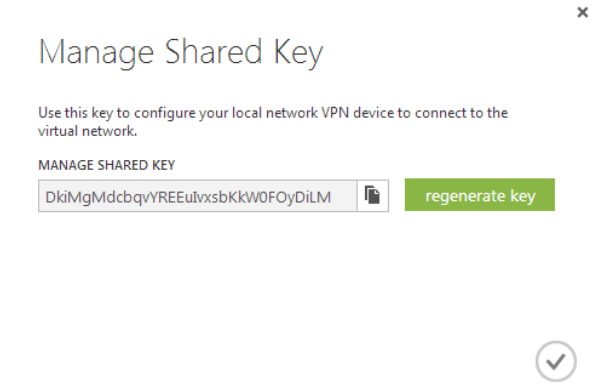
1. In the left pane, click NETWORKS.
2. On the Virtual Network tab, in the Name column, click the virtual network entity.



3. On the DASHBOARD page of the virtual network, copy the Gateway IP Address.



4. For the Pre Shared Key (PSK), at the bottom of the page, click MANAGE KEY.
5. In the MANAGE SHARED KEY dialog box, copy the SHARED KEY.



Configuring the NetScaler Appliance in the Datacenter for the CloudBridge Connector Tunnel

Updated: 2014-04-15

To configure a CloudBridge Connector tunnel between a datacenter and an Azure cloud, perform the following tasks on the NetScaler in the datacenter. You can use either the NetScaler command line or the configuration utility:

- **Create an IPSec profile**. An IPSec profile entity specifies the IPSec protocol parameters, such as IKE version, encryption algorithm, hash algorithm, and PSK, to be used by the IPSec protocol in the CloudBridge Connector tunnel.
- **Create an IP tunnel with IPSec protocol and associate the IPSec profile to it**. An IP tunnel specifies the local IP address (a public SNIP address configured on the NetScaler appliance), remote IP address (the public IP address of the gateway in Azure), protocol (IPSec) used to set up the CloudBridge Connector tunnel, and an IPSec profile entity. The created IP tunnel entity is also called the CloudBridge Connector tunnel entity.
- **Create a PBR rule and associate the IP tunnel to it**. A PBR entity specifies a set of conditions and an IP tunnel (CloudBridge Connector tunnel) entity. The source IP address range and the destination IP range are the conditions for the PBR entity. You must set the source IP address range to specify the datacenter subnet whose traffic is to traverse the tunnel, and the destination IP address range to specify the Azure subnet whose traffic is to traverse the CloudBridge Connector tunnel. Any request packet originated from a client in the subnet on the datacenter and destined to a server in the subnet on the Azure cloud matches the source and destination IP range of the PBR entity. This packet is then considered for CloudBridge Connector tunnel processing and is sent across sent across the CloudBridge Connector tunnel associated with the PBR entity.

The configuration utility combines all these tasks in a single wizard called the CloudBridge Connector wizard.

To create an IPSEC profile by using the NetScaler command line

At the Command prompt, type:

- `add ipsec profile <name> -psk <string> -ikeVersion v1`

To create an IPSEC tunnel and bind the IPSEC profile to it by using the NetScaler command line

At the Command prompt, type:

- add ipTunnel <name> <remote> <remoteSubnetMask> <local> -protocol IPSEC –ipsecProfileName <string>

To create a PBR rule and bind the IPSEC tunnel to it by using the NetScaler command line

At the Command prompt, type:

- add pbr <pbrName> ALLOW –srcIP <subnet-range> -destIP <subnet-range> -ipTunnel <tunnelName>
- apply pbrs

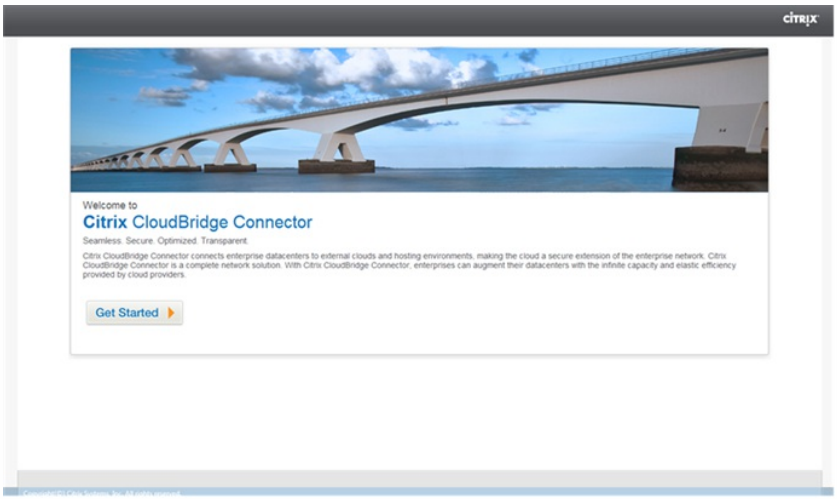
Sample Configuration

The following commands create all settings of NetScaler appliance CB_Appliance-1 used in "Example of CloudBridge Connector Configuration and Data Flow".

```
> add ipsec profile CB_Azure_IPSec_Profile -psk DkiMgMdcbqvYREEuIvxsbKkW0FOyDiLM -ikeVersion v1 –lifetime 31536000
Done
> add iptunnel CB_Azure_Tunnel 168.63.252.133 255.255.255.255 66.165.176.15 –protocol IPSEC –ipsecProfileName CB_Azure_IPSec_Profile
 Done
> add pbr CB_Azure_Pbr-srcIP 10.102.147.0-10.102.147.255 –destIP 10.20.0.0-10.20.255.255 –ipTunnelCB_Azure_Tunnel
 Done
> apply pbrs
Done
```

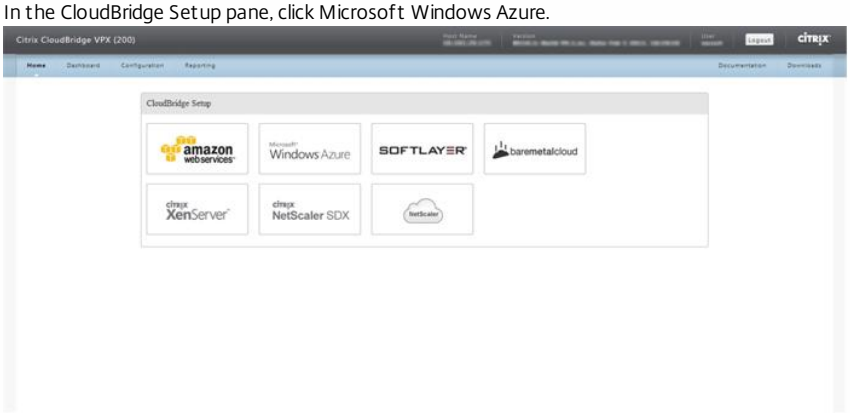
To configure a CloudBridge Connector tunnel in a NetScaler appliance by using the configuration utility

1. Access the configuration utility by using a web browser to connect to the IP address of the NetScaler appliance in the datacenter.
2. Navigate to System > CloudBridge Connector.
3. In the right pane, under Getting Started, click Create/Monitor CloudBridge.
4. Click Get Started.

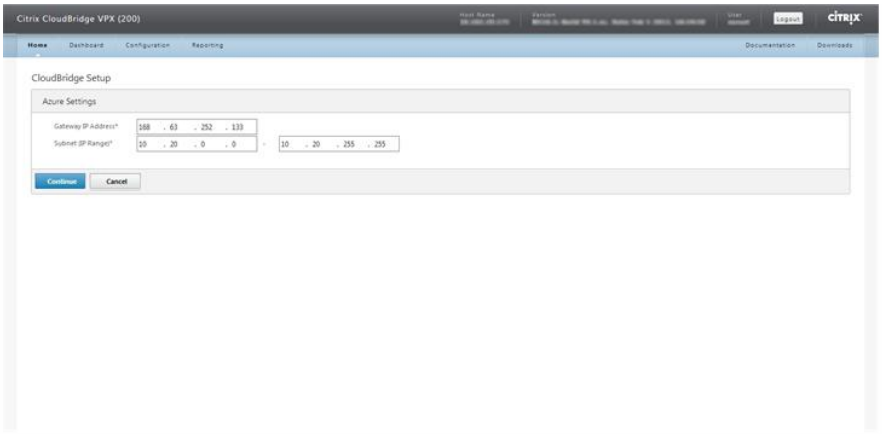


Note: If you already have any CloudBridge Connector tunnel configured on the NetScaler appliance, this screen does not appear, and you are taken to the CloudBridge Connector Setup pane.

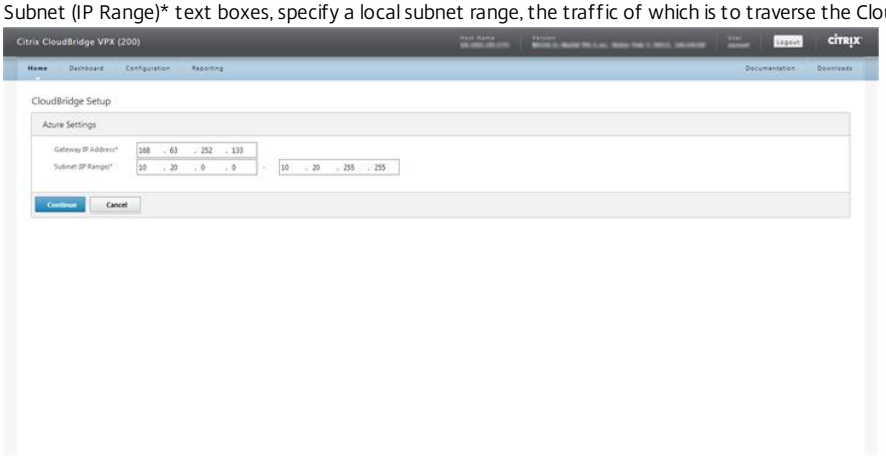
5. In the CloudBridge Setup pane, click Microsoft Windows Azure.



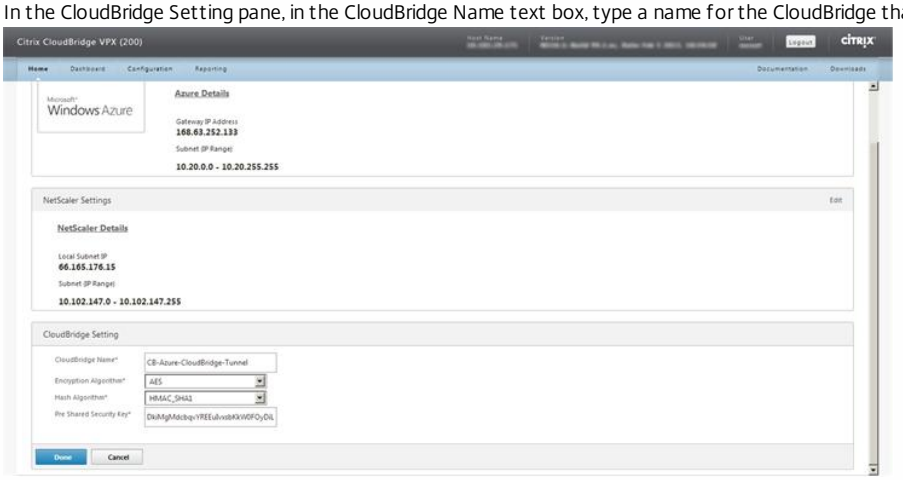
6. In the Azure Settings pane, in the Gateway IP Address* field, type the IP address of the Azure gateway. The CloudBridge Connector tunnel is then set up between the NetScaler appliance and the gateway. In the Subnet (IP Range)* text boxes, specify a subnet range (in Azure cloud), the traffic of which is to traverse the CloudBridge Connector tunnel. Click Continue.



7. In the NetScaler Settings pane, from the Local Subnet IP* drop-down list, select a publicly accessible SNIP address configured on the NetScaler appliance. In Subnet (IP Range)* text boxes, specify a local subnet range, the traffic of which is to traverse the CloudBridge Connector tunnel. Click Continue.



8. In the CloudBridge Setting pane, in the CloudBridge Name text box, type a name for the CloudBridge that you want to create.



9. From the Encryption Algorithm and Hash Algorithm drop-down lists, select the AES and HMAC_SHA1 algorithms, respectively. In the Pre Shared Security Key text box, type the security key.
10. Click Done.

Monitoring the CloudBridge Connector Tunnel

Updated: 2014-04-15

You can view statistics for monitoring the performance of a CloudBridge Connector tunnel between the NetScaler appliance in the datacenter and Microsoft Azure. To view CloudBridge Connector tunnel statistics on the NetScaler appliance, use the configuration utility or the NetScaler command line. To view CloudBridge Connector tunnel statistics in Microsoft Azure, use the Microsoft Windows Azure Management Portal.

Displaying CloudBridge Connector tunnel Statistics in the NetScaler appliance

The following table lists the statistical counters available for monitoring CloudBridge Connector tunnels on a NetScaler appliance.

Statistical counter	Specifies
Bytes Received	Total number of bytes received by the NetScaler appliance through all the configured CloudBridge Connector tunnels since the appliance was last started.
Bytes Sent	Total number of bytes sent by the NetScaler appliance through all the configured CloudBridge Connector tunnels since the appliance was last started.
Packets Received	Total number of packets received by the NetScaler appliance through all the configured CloudBridge Connector tunnels since the appliance was last started.
Packets Sent	Total number of packets sent by the NetScaler appliance through all the configured CloudBridge Connector tunnels since the appliance was last started.

All these counters are reset to 0 when the NetScaler appliance is restarted. They do not increment during the following phases:

- Internet Key Exchange (IKE) authentication (pre-shared key) phase on any configured CloudBridge Connector tunnel.
- IKE Security Association (SA) establishment phase on any configured CloudBridge Connector tunnel.

To display CloudBridge Connector tunnel statistics by using the NetScaler command line

At the command prompt, type:

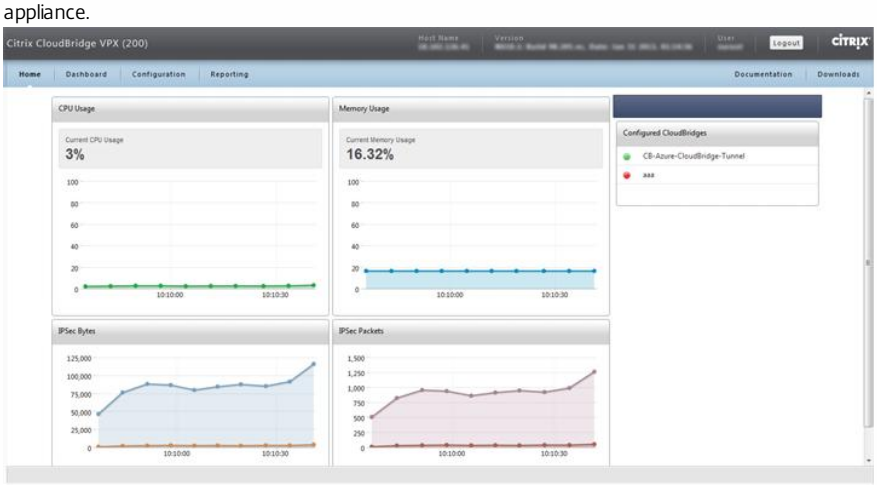
- stat ipsec counters

Example

```
> stat ipsec counters
Secure tunnel(s) summary
            Rate (/s)           Total
Bytes Received  0  2811248
Bytes Sent  0  157460630
Packets Received 0  56787
Packets Sent      0  200910
Done
>
```

To display CloudBridge Connector tunnel statistics by using the Configuration utility

1. Access the configuration utility by using a web browser to connect to the IP address of the NetScaler appliance.
2. On the Home tab, the IPSec Bytes and IPSec Packets charts display the statistics of all the CloudBridge Connector tunnels configured on the NetScaler appliance.



Displaying CloudBridge Connector tunnel Statistics in Microsoft Azure

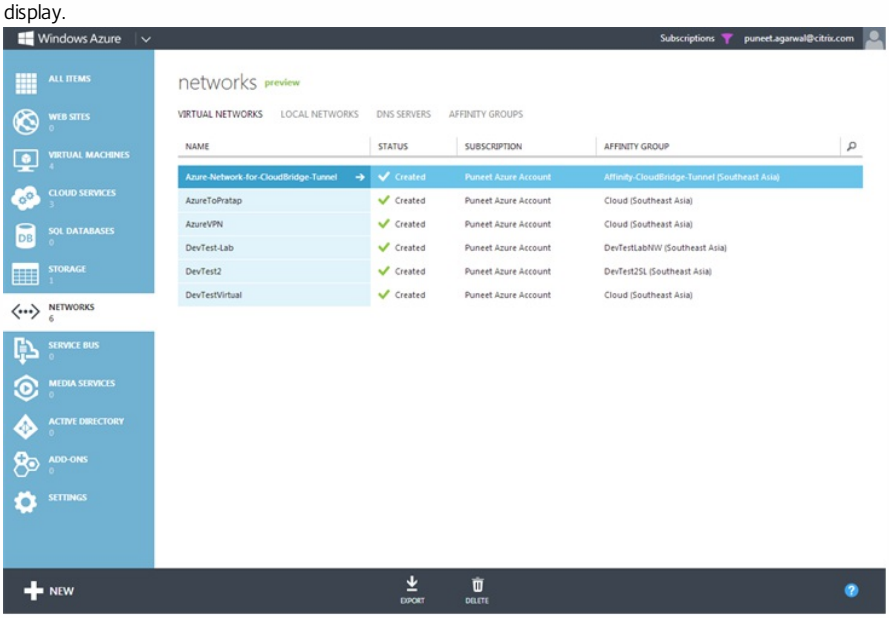
The following table lists the statistical counters available for monitoring CloudBridge Connector tunnels in Microsoft Azure.

Statistical counter	Specifies
DATA IN	Total number of kilobytes received by the Azure gateway through the CloudBridge Connector tunnel since the gateway was created.
DATA OUT	Total number of kilobytes sent by the Azure gateway through the CloudBridge Connector tunnel since the gateway was created.

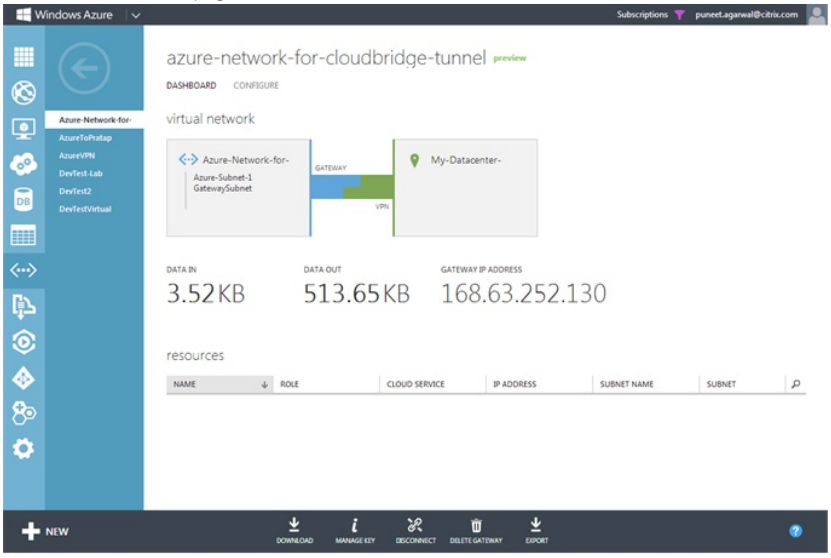
Statistical counter	Specifies

To display CloudBridge Connector tunnel statistics by using the Microsoft Windows Azure Management Portal

1. Log on to the Windows Azure Management Portal (https://manage.windowsazure.com/) by using your Microsoft Azure account credentials.
2. In the left pane, click NETWORKS.
3. On the Virtual Network tab, in the Name column, select the virtual network entity associated with a CloudBridge Connector tunnel whose statistics you want to display.



4. On the DASHBOARD page of the virtual network, view the DATA IN and DATA OUT counters for the CloudBridge Connector tunnel.



Configuring CloudBridge Connector Tunnel between Datacenter and SoftLayer Enterprise Cloud

Jan 31, 2014

The configuration utility includes a wizard that helps you to easily configure a CloudBridge Connector tunnel between a NetScaler appliance in a datacenter and NetScaler VPX instances on the SoftLayer enterprise cloud.

When you use the wizard of the NetScaler appliance in the datacenter, the CloudBridge Connector tunnel configuration created on the NetScaler appliance, is automatically pushed to the other endpoint or peer (the NetScaler VPX on SoftLayer) of the CloudBridge Connector tunnel.

Using the wizard of the NetScaler appliance in the datacenter, you perform the following steps to configure a CloudBridge Connector tunnel.

1. Connect to the Softlayer enterprise cloud by providing the user log on credentials.
2. Select the Citrix XenServer that is running the NetScaler VPX appliance.
3. Select the NetScaler VPX appliance.
4. Provide CloudBridge Connector tunnel parameters to:
   - Configure a GRE Tunnel.
   - Configure IPsec on the GRE tunnel.
   - Create a netbridge, which is a logical representation of the CloudBridge connector, by specifying a name.
   - Bind the GRE Tunnel to the netbridge.

To configure a CloudBridge Connector tunnel by using the configuration utility

1. Log on to the configuration utility of the NetScaler appliance in the datacenter by using your account credentials for the appliance.
2. Navigate to System > CloudBridge Connector .
3. In the right pane, under Getting Started, click Create/Monitor CloudBridge Connector.
4. Click Get Started.
   Note: If you already have any CloudBridge Connector tunnel configured on the NetScaler appliance, this screen does not appear, and you are taken to the CloudBridge Connector Setup pane.
5. In the CloudBridge Connector Setup pane, click Softlayer, and then follow the instructions in the wizard.

High Availability

Mar 19, 2012

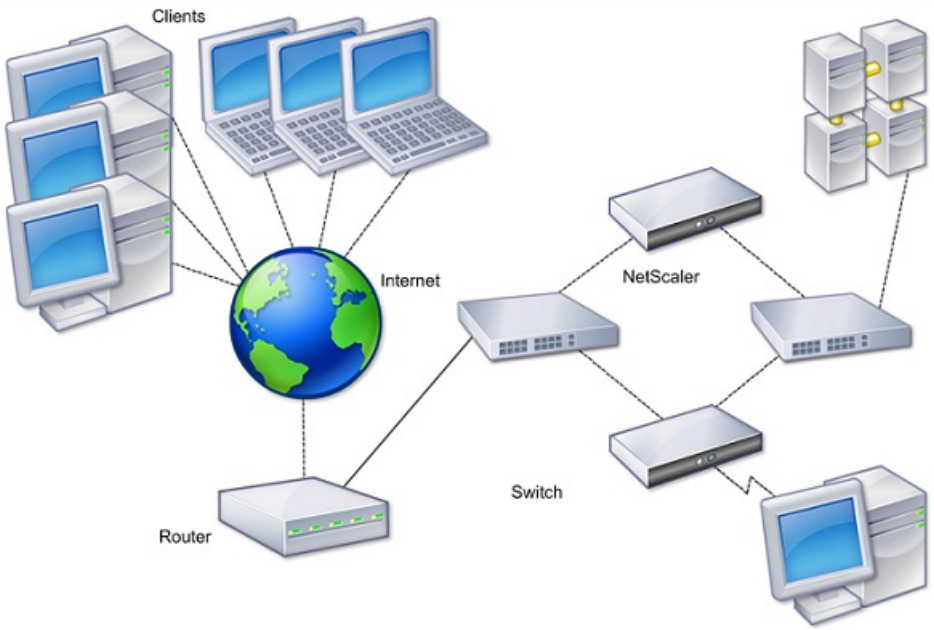
A high availability (HA) deployment of two Citrix® NetScaler® appliances can provide uninterrupted operation in any transaction. With one appliance configured as the primary node and the other as the secondary node, the primary node accepts connections and manages servers while the secondary node monitors the primary. If, for any reason, the primary node is unable to accept connections, the secondary node takes over.

The secondary node monitors the primary by sending periodic messages (often called heartbeat messages or health checks) to determine whether the primary node is accepting connections. If a health check fails, the secondary node retries the connection for a specified period, after which it determines that the primary node is not functioning normally. The secondary node then takes over for the primary (a process called failover).

After a failover, all clients must reestablish their connections to the managed servers, but the session persistence rules are maintained as they were before the failover.

With Web server logging persistence enabled, no log data is lost due to the failover. For logging persistence to be enabled, the log server configuration must carry entries for both systems in the log.conf file.

The following figure shows a network configuration with an HA pair.
Figure 1. NetScaler Appliances in a High Availability Configuration



To configure HA, you might want to begin by creating a basic setup, with both nodes in the same subnet. You can then customize the intervals at which the nodes communicate health-check information, the process by which nodes maintain synchronization, and the propagation of commands from the primary to the secondary. You can configure fail-safe mode to prevent a situation in which neither node is primary. If your environment includes devices that do not accept NetScaler gratuitous ARP messages, you should configure virtual MAC addresses. When you are ready for a more complex configuration, you can configure HA nodes in different subnets.

To improve the reliability of your HA setup, you can configure route monitors and create redundant links. In some situations, such as when troubleshooting or performing maintenance tasks, you might want to force a node to fail over (assign primary

status to the other node), or you might want to force the secondary node to stay secondary or the primary node to stay primary.

Considerations for a High Availability Setup

Mar 19, 2012

Note the following requirements for configuring systems in an HA setup:

- In an HA configuration, the primary and secondary NetScaler appliances should be of the same model. Different NetScaler models are not supported in an HA pair (for example, you cannot configure a 10010 model and a 7000 model as an HA pair).
- In an HA setup, both nodes must run the same version of NetScaler, for example, nCore/nCore or classic/classic. If the nodes are running NetScaler classic and you want to migrate to NetScaler nCore of the same NetScaler release, prop and sync are not supported during the migration process. Once migration is complete, prop and sync are auto-enabled. The same applies if you migrate from NetScaler nCore to NetScaler classic.
- Entries in the configuration file (ns.conf) on both the primary and the secondary system must match, with the following exceptions:
  - The primary and the secondary systems must each be configured with their own unique NetScaler IP addresses (NSIPs.)
  - In an HA pair, the node ID and associated IP address of one node must point to the other node. For example, if you have nodes NS1 and NS2, you must configure NS1 with a unique node ID and the IP address of NS2, and you must configure NS2 with a unique node ID and the IP address of NS1.
- If you create a configuration file on either node by using a method that does not go directly through the GUI or the CLI (for example, importing SSL certificates, or changing to startup scripts), you must copy the configuration file to the other node or create an identical file on that node.
- Initially, all NetScaler appliances are configured with the same RPC node password. RPC nodes are internal system entities used for system-to-system communication of configuration and session information. For security, you should change the default RPC node passwords.

  One RPC node exists on each NetScaler. This node stores the password, which is checked against the password provided by the contacting system. To communicate with other systems, each NetScaler requires knowledge of those systems, including how to authenticate on those systems. RPC nodes maintain this information, which includes the IP addresses of the other systems, and the passwords they require for authentication.

  RPC nodes are implicitly created when adding a node or adding a Global Server Load Balancing (GSLB) site. You cannot create or delete RPC nodes manually.

  Note: If the NetScaler appliances in a high availability setup are configured in one-arm mode, you must disable all system interfaces except the one connected to the switch or hub.
- For an IPv6 HA configuration, the following considerations apply:
  - You must install the IPv6PT license on both NetScaler appliances.
  - After installing the IPv6PT license, enable the IPv6 feature by using the configuration utility or the command line interface.
  - Both NetScaler appliances require a global NSIP IPv6 address. In addition, network entities (for example, switches and routers) between the two nodes must support IPv6.

Configuring High Availability

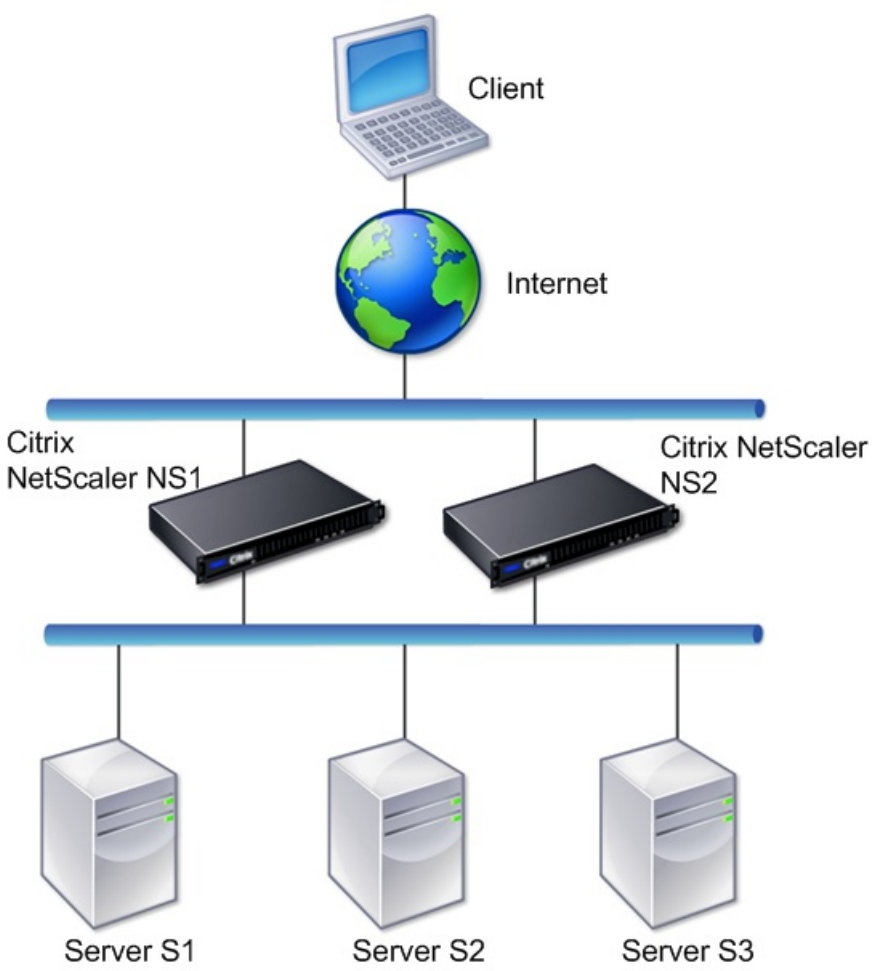
Mar 19, 2012

To set up a high availability configuration, you create two nodes, each of which defines the other's NetScaler IP (NSIP) address as a remote node. Begin by logging on to one of the two NetScaler appliances that you want to configure for high availability, and add a node. Specify the other appliance's NetScaler IP (NSIP) address as the address of the new node. Then, log on to the other appliance and add a node that has the NSIP address of the first appliance. An algorithm determines which node becomes primary and which becomes secondary.

Note: The configuration utility provides an option that avoids having to log on to the second appliance.

The following figure shows a simple HA setup, in which both nodes are in same subnet.

Figure 1. Two NetScaler Appliances Connected in a High Availability Configuration



Adding a Remote Node

To add a remote NetScaler appliance as a node in a high availability setup, you specify a unique node ID and the appliance's NSIP. The maximum number of node IDs in an HA setup is 64. When you add an HA node, you must disable the HA monitor for each interface that is not connected or not being used for traffic. For CLI users, this is a separate procedure.

Note: To ensure that each node in the high availability configuration has the same settings, you should synchronize your SSL certificates, startup scripts, and other configuration files with those on the primary node.

To add a node by using the command line interface

At the command prompt, type:

- add ha node <id> <IPAddress>
- show ha node

Example

```
> add ha node 3 1000:0000:0000:0000:0005:0600:700a:888b
```

To disable an HA monitor by using the command line interface

At the command prompt, type:

- set interface <ifNum> [-haMonitor (**ON** | **OFF**)]
- show interface <ifNum>

Example

```
> set interface 1/3 -haMonitor OFF
 Done
```

To add a remote node by using the configuration utility

Navigate to System > High Availability and, on the Nodes tab, add a new remote node, or edit an existing node.

Disabling or Enabling a Node

Updated: 2013-08-28

You can disable or enable only a secondary node. When you disable a secondary node, it stops sending heartbeat messages to the primary node, and therefore the primary node can no longer check the status of the secondary. When you enable a node, the node takes part in the high availability configuration.

To disable or enable a node by using the command line interface

At the command prompt, type one of the following commands:

- set ha node -hastatus DISABLED
- set ha node -hastatus ENABLED

To disable or enable a node by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, open the node.
2. In the High Availability Status list, select ENABLED (Actively Participate in HA) or DISABLED (Do not participate in HA).

Removing a Node

Updated: 2013-08-28

If you remove a node, the nodes are no longer in high availability configuration.

To remove a node by using the command line interface

At the command prompt, type:

rm ha node <id>

Example

```
> rm ha node 2
  Done
```

To remove a node by using the configuration utility

Navigate to System > High Availability and, on the Nodes tab, delete the node.

Note: You can use the Network Visualizer to view the NetScaler appliances that are configured as a high availability (HA) pair and perform high availability configuration tasks. For more information, see "Using the Network Visualizer."

Configuring the Communication Intervals

Mar 19, 2012

The hello interval is the interval at which the heartbeat messages are sent to the peer node. The dead interval is the time interval after which the peer node is marked DOWN if heartbeat packets are not received. The heartbeat messages are UDP packets sent to port 3003 of the other node in an HA pair.

To set the hello and dead intervals by using the command line interface

At the command prompt, type:

- set HA node [-helloInterval <msecs>] [-deadInterval <secs>]
- show HA node <id>

To set the hello and dead intervals by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, open the node.
2. Set the following parameters:
   - Hello Interval (msecs)
   - Dead Interval (secs)

Configuring Synchronization

Mar 19, 2012

Synchronization is a process of duplicating the configuration of the primary node on the secondary node. The purpose of synchronization is to ensure that there is no loss of configuration information between the primary and the secondary nodes, regardless of the number of failovers that occur. Synchronization uses port 3010.

Synchronization is triggered by either of the following circumstances:

- The secondary node in an HA setup comes up after a restart.
- The primary node becomes secondary after a failover.

Automatic synchronization is enabled by default. You can also force synchronization.

Disabling or Enabling Synchronization

Updated: 2013-08-28

Automatic HA synchronization is enabled by default on each node in an HA pair. You can enable or disable it on either node.

To disable or enable automatic synchronization by using the command line interface

At the command prompt, type:

- set HA node -haSync DISABLED
- set HA node -haSync ENABLED

To disable or enable synchronization by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, open the node.
2. Under HA Synchronization, clear or select the Secondary node will fetch the configuration from Primary option.

Forcing the Secondary Node to Synchronize with the Primary Node

Updated: 2013-08-28

In addition to automatic synchronization, the NetScaler supports forced synchronization. You can force the synchronization from either the primary or the secondary node. When you force synchronization from the secondary node, it starts synchronizing its configuration with the primary node.

However, if synchronization is already in progress, forced synchronization fails and the system displays a warning. Forced synchronization also fails in any of the following circumstances:

- You force synchronization on a standalone system.
- The secondary node is disabled.
- HA synchronization is disabled on the secondary node.

To force synchronization by using the command line interface

At the command prompt, type:

force HA sync

To force synchronization by using the configuration utility

1. Navigate to System > High Availability.
2. On the Nodes tab, in the Action list, click Force Synchronization.

Synchronizing Configuration Files in a High Availability Setup

Sep 05, 2013

In a high availability setup, you can synchronize various configuration files from the primary node to the secondary node.

To perform the synchronization, you can use the command line interface or the configuration utility at either the primary or the secondary node. Files located on the secondary that are specific to the secondary (not present on the primary) are not deleted during the synchronization.

To synchronize files in a high availability setup by using the command line interface

At the command prompt, type:

sync HA files <mode>

Example

```
> sync HA files all
 Done
```

To synchronize files in a high availability setup by using the configuration utility

Navigate to System > Diagnostics and, in the Utilities group, click Start HA files synchronization.

Configuring Command Propagation

Aug 28, 2013

In an HA setup, any command issued on the primary node propagates automatically to, and is executed on, the secondary before it is executed on the primary. If command propagation fails, or if command execution fails on the secondary, the primary node executes the command and logs an error. Command propagation uses port 3010.

In an HA pair configuration, command propagation is enabled by default on both the primary and secondary nodes. You can enable or disable command propagation on either node in an HA pair. If you disable command propagation on the primary node, commands are not propagated to the secondary node. If you disable command propagation on the secondary node, commands propagated from the primary are not executed on the secondary node.

Note: After reenabling propagation, remember to force synchronization.
If synchronization occurs while you are disabling propagation, any configuration-related changes that you make before the disabling of propagation takes effect are synchronized with the secondary node. This is also true for cases where propagation is disabled while synchronization is in progress.

To disable or enable command propagation by using the command line interface

At the command prompt, type:

- set HA node -haProp DISABLED
- set HA node -haProp ENABLED

To disable or enable command propagation by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, open the node.
2. Clear or select the Primary node will propagate configuration to the Secondary option.

Configuring Fail-Safe Mode

Aug 28, 2013

In an HA configuration, fail-safe mode ensures that one node is always primary when both nodes fail the health check. This is to ensure that when a node is only partially available, backup methods are enabled to handle traffic as best as possible. The HA fail-safe mode is configured independently on each node.

The following table shows some of the fail-safe cases. The NOT_UP state means that the node failed the health check yet it is partially available. The UP state means that the node passed the health check.

Table 1. Fail-Safe Mode Cases

Node A (Primary) Health State	Node B (Secondary) Health State	Default HA Behavior	Fail-Safe Enabled HA Behavior	Description
NOT_UP(failed last)	NOT_UP (failed first)	A (Secondary), B (Secondary)	A (Primary), B (Secondary)	If both nodes fail, one after the other, the node that was the last primary remains primary.
NOT_UP (failed first)	NOT_UP(failed last)	A (Secondary), B (Secondary)	A(Secondary), B(Primary)	If both nodes fail, one after the other, the node that was the last primary remains primary.
UP	UP	A (Primary), B (Secondary)	A (Primary), B (Secondary)	If both nodes pass the health check, no change in behavior with fail-safe enabled.
UP	NOT_UP	A(Primary), B(Secondary)	A (Primary), B (Secondary)	If only the secondary node fails, no change in behavior with fail-safe enabled.
NOT_UP	UP	A(Secondary), B(Primary)	A(Secondary), B(Primary)	If only the primary fails, no change in behavior with fail-safe enabled.
NOT_UP	UP (STAYSECONDARY)	A (Secondary), B (Secondary)	A (Primary), B (Secondary)	If the secondary is configured as STAYSECONDARY, the primary remains primary even if it fails.

To enable fail-safe mode by using the command line interface

At the command prompt, type:

set HA node [-failSafe (**ON** | **OFF**)]

Example

```
set ha node -failsafe ON
```

To enable fail-safe mode by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, open the node.
2. Under Fail-Safe Mode, select the Maintain one Primary node even when both nodes are unhealthy option.

Configuring Virtual MAC Addresses

Mar 19, 2012

A Virtual MAC address (VMAC) is a floating entity shared by the primary and the secondary nodes in an HA setup.

In an HA setup, the primary node owns all of the floating IP addresses, such as the MIPs, SNIPs, and VIPs. The primary node responds to Address Resolution Protocol (ARP) requests for these IP addresses with its own MAC address. As a result, the ARP table of an external device (for example, an upstream router) is updated with the floating IP address and the primary node's MAC address.

When a failover occurs, the secondary node takes over as the new primary node. It then uses Gratuitous ARP (GARP) to advertise the floating IP addresses that it acquired from the primary. However, the MAC address that the new primary advertises is the MAC address of its own interface.

Some devices (notably a few routers) do not accept the GARP messages generated by the NetScaler appliance. As a result, some external devices retain the old IP to MAC mapping advertised by the old primary node. This can result in a site going down.

You can overcome this problem by configuring a VMAC on both nodes of an HA pair. Both nodes then possess identical MAC addresses. Therefore, when failover occurs, the MAC address of the secondary node remains unchanged, and the ARP tables on the external devices do not need to be updated.

To create a VMAC, you need to first create a Virtual Router ID (VRID) and bind it to an interface. (In an HA setup, you need to bind the VRID to the interfaces on both nodes.) Once the VRID is bound to an interface, the system generates a VMAC with the VRID as the last octet.

This section includes the following details:

- Configuring IPv4 VMACs
- Configuring IPv6 VMAC6s

Configuring IPv4 VMACs

When you create a IPv4 VMAC address and bind it to a interface, any IPv4 packet sent from the interface uses the VMAC address that is bound to the interface. If there is no IPv4 VMAC bound to an interface, the interface's physical MAC address is used.

The generic VMAC is of the form 00:00:5e:00:01:<VRID>. For example, if you create a VRID with a value of 60 and bind it to an interface, the resulting VMAC is 00:00:5e:00:01:3c, where 3c is the hex representation of the VRID. You can create 255 VRIDs with values from 1 to 255.

Creating or Modifying an IPv4 VMAC

Updated: 2013-08-28

You create an IPv4 virtual MAC by assigning it a virtual router ID. You can then you bind the VMAC to an interface. You cannot bind multiple VRIDs to the same interface. To verify the VMAC configuration, you should display and examine the VMACs and the interfaces bound to the VMACs.
To add a VMAC by using the command line interface

At the command prompt, type:

- add vrID <id>
- bind vrid <id> -ifnum <interface_name>
- show vrID

Example

```
> add vrID 100
 Done
> bind vrid 100 -ifnum 1/1 1/2 1/3
 Done
```

To unbind interfaces from a VMAC by using the command line interface

At the command prompt, type:

- unbind vrid <id> -ifnum <interface_name>
- show vrID

To configure a VMAC by using the configuration utility

Navigate to System > Network > VMAC and, on the VMAC tab, add a new VMAC, or edit an existing VMAC.

Removing an IPv4 VMAC

Updated: 2013-08-28

To remove an IPv4 virtual MAC, you delete its virtual router ID.
To remove an IPv4 VMAC by using the command line interface

At the command prompt, type:
rm vrid <id>
Example

```
rm vrid 100s
```

To remove an IPv4 VMAC by using the configuration utility

Navigate to System > Network > VMAC and, on the VMAC tab, delete the IPv4 VMAC.

Configuring IPv6 VMAC6s

The NetScaler supports VMAC6 for IPv6 packets. You can bind any interface to a VMAC6, even if an IPv4 VMAC is bound to the interface. Any IPv6 packet sent from the interface uses the VMAC6 bound to that interface. If there is no VMAC6 bound to an interface, an IPv6 packet uses the physical MAC.

Creating or Modifying a VMAC6

Updated: 2013-08-28

You create an IPv6 virtual MAC by assigning it an IPv6 virtual router ID. You can then you bind the VMAC to an interface. You cannot bind multiple IPv6 VRIDs to an interface. To verify the VMAC6 configuration, you should display and examine the VMAC6s and the interfaces bound to the VMAC6s.
To add a VMAC6 by using the command line interface

At the command prompt, type:

- add vrID6 <id>
- bind vrID6 <id> -ifnum <interface_name>
- show vrID6

Example

```
> add vrID6 100
Done
> bind vrID6 100 -ifnum 1/1 1/2 1/3
 Done
```

To unbind interfaces from a VMAC6 by using the command line interface

At the command prompt, type:

- unbind vrID6 <id> -ifnum <interface_name>
- show vrID6

To configure a VMAC6 by using the configuration utility

Navigate to System > Network > VMAC and, on the VMAC6 tab, add a new VMAC6, or edit an existing VMAC6.

Removing a VMAC6

Updated: 2013-08-28

To remove an IPv4 virtual MAC, you delete its virtual router ID.
To remove a VMAC6 by using the command line interface

At the command prompt, type:
rm vrid6 <id>
Example

```
rm vrid6 100s
```

To remove a VMAC6 by using the configuration utility

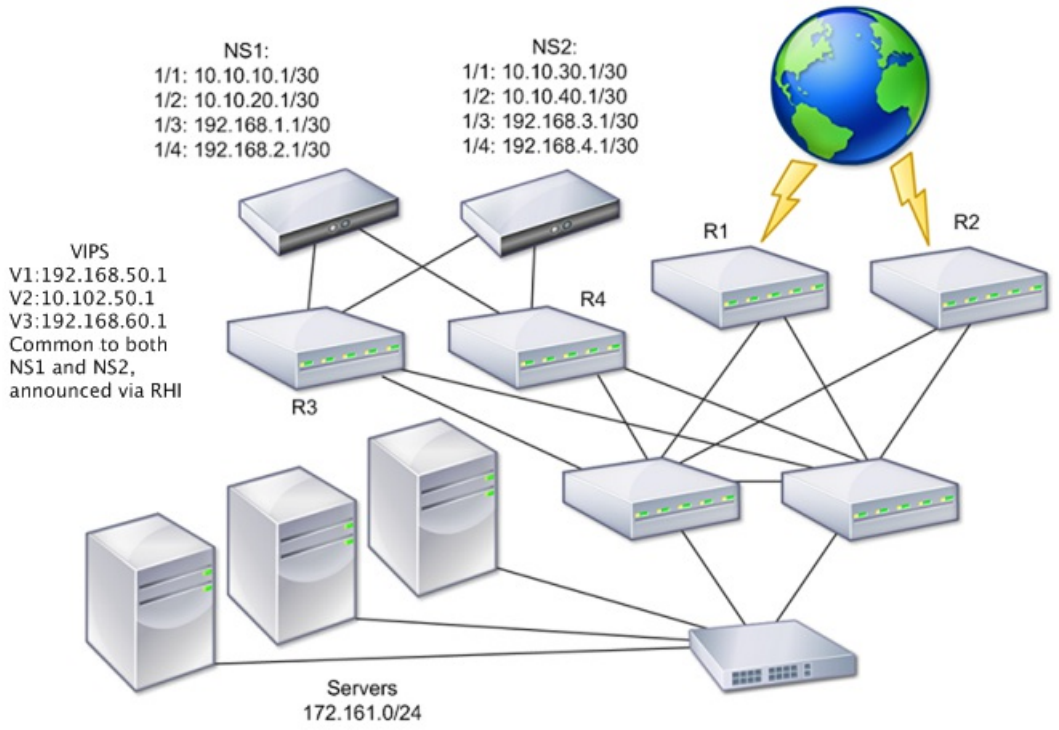
Navigate to System > Network > VMAC and, on the VMAC6 tab, delete the virtual router ID.

Configuring High Availability Nodes in Different Subnets

Mar 19, 2012

The following figure shows an HA deployment with the two systems located in different subnets:

Figure 1. High Availability over a Routed Network



In the figure, the systems NS1 and NS2 are connected to two separate routers, R3 and R4, on two different subnets. The NetScaler appliances exchange heartbeat packets through the routers. This configuration could be expanded to accommodate deployments involving any number of interfaces.

Note: If you use static routing on your network, you must add static routes between all the systems to ensure that heartbeat packets are sent and received successfully. (If you use dynamic routing on your systems, static routes are unnecessary.)

If the nodes in an HA pair reside on two separate networks, the primary and secondary node must have independent network configurations. This means that nodes on different networks cannot share entities such as MIPs, SNIPs, VLANs, and routes. This type of configuration, where the nodes in an HA pair have different configurable parameters, is known as Independent Network Configuration (INC) or Symmetric Network Configuration (SNC).

The following table summarizes the configurable entities and options for an INC, and shows how they must be set on each node.

Table 1. Behavior of NetScaler Entities and Options in an Independent Network Configuration

NetScaler entities	Options

NetScaler entities	Options
IPs (NSIP/MIP/SNIPs)	Node-specific. Active only on that node.
VIPs	Floating.
VLANs	Node-specific. Active only on that node.
Routes	Node-specific. Active only on that node. Link load balancing routes are floating.
ACLs	Floating (Common). Active on both nodes.
Dynamic routing	Node-specific. Active only on that node. The secondary node should also run the routing protocols and peer with upstream routers.
L2 mode	Floating (Common). Active on both nodes.
L3 mode	Floating (Common). Active on both nodes.
Reverse NAT (RNAT)	Node-specific. RNAT with VIP, because NATIP is floating.

As in configuring HA nodes in the same subnet, to configure HA nodes in different subnets, you log on to each of the two NetScaler appliances and add a remote node representing the other appliance.

Adding a Remote Node

When two nodes of an HA pair reside on different subnets, each node must have a different network configuration. Therefore, to configure two independent systems to function as an HA pair, you must specify INC mode during the configuration process.

When you add an HA node, you must disable the HA monitor for each interface that is not connected or not being used for traffic. For CLI users, this is a separate procedure.

To add a node by using the command line interface

At the command prompt, type:

- add ha node <id> <IPAddress> -inc ENABLED
- show ha node

Example

```
> add ha node 3 10.102.29.170 -inc ENABLED
 Done
> add ha node 3 1000:0000:0000:0000:0005:0600:700a:888b
```

Done

To disable an HA monitor by using the command line interface

At the command prompt, type:

- set interface <ifNum> [-haMonitor (**ON** | **OFF**)]
- show interface <ifNum>

Example

```
> set interface 1/3 -haMonitor OFF
 Done
```

To add a remote node by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, add a new remote node.
2. Make sure to select the Turn off HA monitor on interfaces/channels that are down and Turn on INC (Independent Network Configuration) mode on self mode options.

Removing a Node

Updated: 2013-08-28

If you remove a node, the nodes are no longer in high availability configuration.

To remove a node by using the command line interface

At the command prompt, type:

rm ha node <id>

Example

```
> rm ha node 2
 Done
```

To remove a node by using the configuration utility

Navigate to System > High Availability and, on the Nodes tab, delete the node.

Note: You can use the Network Visualizer to view the NetScaler appliances that are configured as a high availability (HA) pair and perform high availability configuration tasks. For more information, see "Using the Network Visualizer."

Configuring Route Monitors

Sep 06, 2013

You can use route monitors to make the HA state dependent on the internal routing table, whether or not the table contains any dynamically learned or static routes. In an HA configuration, a route monitor on each node watches the internal routing table to make sure that a route entry for reaching a particular network is always present. If the route entry is not present, the state of the route monitor changes to DOWN.

When a NetScaler appliance has only static routes for reaching a network, and you want to create a route monitor for the network, you must enable monitored static routes (MSR) for the static routes. MSR removes unreachable static routes from the internal routing table. If MSR is disabled on static routes, an unreachable static route can remain in the internal routing table, defeating the purpose of having the route monitor.

Route Monitors are supported both in non-INC and INC mode.

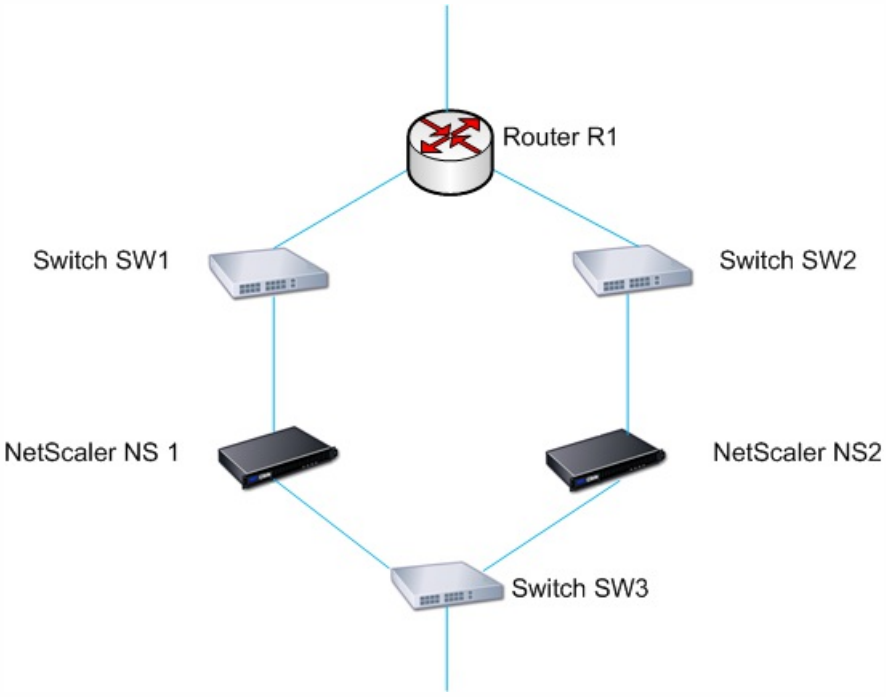
Route Monitors in HA in non-INC mode	**Route Monitors in HA in INC mode**
Route monitors are propagated by nodes and exchanged during synchronization.	Route monitors are neither propagated by nodes nor exchanged during synchronization.
Route monitors are active only in the current primary node.	Route monitors are active on both the primary and the secondary node.
The NetScaler appliance always displays the state of a route monitor as UP irrespective of the whether the route entry is present or not in the internal routing table.	The NetScaler appliance displays the state of the route monitor as DOWN if the corresponding route entry is not present in the internal routing table.
A route monitor starts monitoring its route after 180 seconds in the following cases [This is done to allow dynamic routes to get learnt, which may take 180 secs]: • reboot • failover • set route6 command for v6 routes • set route msr enable/disable command for v4 routes. • adding a new route monitor	-

Route monitors are useful in a non-INC mode HA configuration where you want the non-reachability of a gateway from a primary node to be one of the conditions for HA failover.

Consider an example of a non-Inc mode HA setup in a two-arm topology that has NetScaler appliances NS1 and NS2 in the same subnet, with router R1 and switches SW1, SW2, and SW3.

Because R1 is the only router in this setup, you want the HA setup to failover whenever R1 is not reachable from the current primary node. You can configure a route monitor (say, RM1 and RM2, respectively) on each of the nodes to monitor the reachability of R1 from that node.

Figure 1.



With NS1 as the current primary node, the execution flow is as follows:

1. Route monitor RM1 on NS1 monitors NS1's internal routing table for the presence of a route entry for router R1. NS1 and NS2 exchange heartbeat messages through switch SW1 or SW3 at regular intervals.
2. If switch SW1 goes down, the routing protocol on NS1 detects that R1 is not reachable and therefore removes the route entry for R1 from the internal routing table. NS1 and NS2 exchanges heartbeat messages through switch SW3 at regular intervals.
3. Detecting that the route entry for R1 is not present in the internal routing table, RM1 initiates a failover. If route to R1 is down from both NS1 and NS2, failover happens every 180 seconds till one of the appliances is able to reach R1 and restore the connectivity.

Adding a Route Monitor to a High Availability Node

A single procedure creates a route monitor and binds it to an HA node.

To add a route monitor by using the command line interface

At the command prompt, type:

- bind HA node <id> (-routeMonitor <ip_addr|ipv6_addr> [<netmask>])
- show HA node

Example

```
> bind HA node 0 -routeMonitor 10.102.71.0 255.255.255.0
 Done
> bind HA node 0 -routeMonitor 1000:0000:0000:0000:0005:0600:700a:888b
 Done
```

To add a route monitor by using the configuration utility

Navigate to System > High Availability and, on the Route Monitors tab, click Configure.

Removing Route Monitors

Updated: 2013-08-28

To remove a route monitor by using the command line interface

At the command prompt, type:

- unbind HA node <id> (-routeMonitor <ip_addr|ipv6_addr> [<netmask>])
- show ha node

Example

```
unbind HA node 3 -routeMonitor 10.102.71.0 255.255.255.0
unbind HA node 3 -routeMonitor 1000:0000:0000:0000:0005:0600:700a:888b
```

To remove a route monitor by using the configuration utility

Navigate to System > High Availability and, on the Route Monitors tab, delete the route monitor.

Limiting Failovers Caused by Route Monitors in non-INC mode

Aug 28, 2013

In an HA configuration in non-INC mode, if route monitors fail on both nodes, failover happens every 180 seconds until one of the nodes is able to reach all of the routes monitored by the respective route monitors.

However, for a node, you can limit the number of failovers for a given interval by setting the Maximum Number of Flips and Maximum Flip Time parameters on the nodes. When either limit is reached, no more failovers occur, and the node is assigned as primary even if any route monitor fails on that node. If the node is then able to reach all of the monitored routes, the next monitor failure triggers resetting of the Maximum Number of Flips and Maximum Flip Time parameters on the node and starting the time specified in the Maximum Flip Time parameter.

These parameters are set independently on each node and therefore are neither propagated nor synchronized.

Parameters for limiting the number of failovers

Maximum Number of Flips (maxFlips)

Maximum number of failovers allowed, within the Maximum Flip Time interval, for the node in HA in non INC mode, if the failovers are caused by route-monitor failure.

Maximum Flip Time (maxFlipTime)

Amount of time, in seconds, during which failovers resulting from route-monitor failure are allowed for the node in HA in non INC mode.

To limit the number of failovers by using the command line interface

At the command prompt, type:

- set HA node [-maxFlips < positive_integer>] [-maxFlipTime <positive_integer>]
- show HA node [< id>]

Example

```
> set ha node -maxFlips 30 -maxFlipTime 60
 Done
> sh ha node
1) Node ID: 0
 IP: 10.102.169.82 (NS)
 Node State: UP
 Master State: Primary
 Fail-Safe Mode: OFF
 INC State: DISABLED
 Sync State: ENABLED
 Propagation: ENABLED
 Enabled Interfaces : 1/1
 Disabled Interfaces : None
 HA MON ON Interfaces : 1/1
```

```
 Interfaces on which heartbeats are not seen :None
 Interfaces causing Partial Failure:None
 SSL Card Status: NOT PRESENT
 Hello Interval: 200 msecs
 Dead Interval: 3 secs
 Node in this Master State for: 0:4:24:1
 (days:hrs:min:sec)
2) Node ID: 1
 IP: 10.102.169.81
 Node State: UP
 Master State: Secondary
 Fail-Safe Mode: OFF
 INC State: DISABLED
 Sync State: SUCCESS
 Propagation: ENABLED
 Enabled Interfaces : 1/1
 Disabled Interfaces : None
 HA MON ON Interfaces : 1/1
 Interfaces on which heartbeats are not seen : None
 Interfaces causing Partial Failure: None
 SSL Card Status: NOT PRESENT

Local node information:
 Configured/Completed Flips: 30/0
 Configured Flip Time: 60
 Critical Interfaces: 1/1

Done
```

To limit the number of failovers by using the configuration utility

1. Navigate to System > High Availability and, on the Nodes tab, open the local node.
2. Set the following parameters:
   - Maximum Number of Flips
   - Maximum Flip Time

Configuring Failover Interface Set

Mar 19, 2012

A Failover Interface Set (FIS) is a logical group of interfaces. In an HA configuration, using a FIS is a way to prevent failover by grouping interfaces so that, when one interface fails, other functioning interfaces are still available. A FIS can also be configured for the nodes of a NetScaler cluster.

HA MON interfaces that are not bound to an FIS are known as critical interfaces (CI) because if any of them fails, failover is triggered.

> **Note**
>
> An FIS does not create an active and standby Interfaces or channels. It also does not prevent bridging loops when connecting to links to the same VLAN.

Creating or Modifying an FIS

To add an FIS and bind interfaces to it by using the command line interface

At the command prompt, type:

- add fis <name>
- bind fis <name> <ifnum> ...
- show fis <name>

Example

```
> add fis fis1
 Done
> bind fis fis1 1/3 1/5
 Done
```

An unbound interface becomes a critical interface (CI) if it is enabled and HA MON is on.

To unbind an interface from an FIS by using the command line interface

At the command prompt, type:

- unbind fis <name> <ifnum> ...
- show fis <name>

Example

```
> unbind fis fis1 1/3
 Done
```

To configure an FIS by using the configuration utility

Navigate to System > High Availability and, on the Failover Interface Set tab, add a new FIS, or edit an existing FIS.

Removing an FIS

Updated: 2013-08-28

When the FIS is removed, its interfaces are marked as critical interfaces.

To remove an FIS by using the command line interface

At the command prompt, type:

rm fis <name>

Example

```
> rm fis fis1
  Done
```

To remove an FIS by using the configuration utility

Navigate to System > High Availability and, on the Failover Interface Set tab, delete the FIS.

Understanding the Causes of Failover

Sep 30, 2013

The following events can cause failover in an HA configuration:

1. If the secondary node does not receive a heartbeat packet from the primary for a period of time that exceeds the dead interval set on the secondary. (See Note: 1.)
2. The primary node experiences a hardware failure of its SSL card.
3. The primary node does not receive any heartbeat packets on its network interfaces for three seconds.
4. On the primary node, a network interface that is not part of a Failover Interface Set (FIS) or a Link Aggregation (LA) channel and has the HA Monitor (HAMON) enabled, fails. (See Note: 2.)
5. On the primary node, all interfaces in an FIS fail. (See Note: 2.)
6. On the primary node, an LA channel with HAMON enabled fails. (See Note: 2.)
7. On the primary node, all interfaces fail (see Note: 2). In this case, failover occurs regardless of the HAMON configuration.
8. On the primary node, all interfaces are manually disabled. In this case, failover occurs regardless of the HAMON configuration.
9. You force a failover by issuing the force failover command on either node.
10. A route monitor that is bound to the primary node goes DOWN.

    Note: 1. For more information about setting the dead interval, see Configuring the Communication Intervals. Possible causes for a node not receiving heartbeat packets from a peer node include:

    - A network configuration problem prevents heartbeats from traversing the network between the HA nodes.
    - The peer node experiences a hardware or software failure that causes it to freeze (hang), reboot, or otherwise stop processing and forwarding heartbeat packets.

    Note: 2. In this case, fail means that the interface was enabled but goes to the DOWN state, as can be seen from the show interface command or from the configuration utility. Possible causes for an enabled interface to be in the DOWN state are LINK DOWN and TXSTALL.

Forcing a Node to Fail Over

Mar 19, 2012

You might want to force a failover if, for example, you need to replace or upgrade the primary node. You can force failover from either the primary or the secondary node. A forced failover is not propagated or synchronized. To view the synchronization status after a forced failover, you can view the status of the node.

A forced failover fails in any of the following circumstances:

- You force failover on a standalone system.
- The secondary node is disabled.
- The secondary node is configured to remain secondary.

The NetScaler appliance displays a warning message if it detects a potential issue when you run the force failover command. The message includes the information that triggered the warning, and requests confirmation before proceeding.

You can force a failover on a primary node, secondary node, and when nodes are in listen mode.

- **Forcing Failover on the Primary Node**.
  If you force failover on the primary node, the primary becomes the secondary and the secondary becomes the primary. Forced failover is possible only when the primary node can determine that the secondary node is UP.

  If the secondary node is DOWN, the force failover command returns the following error message: "Operation not possible due to invalid peer state. Rectify and retry."

  If the secondary system is in the claiming state or inactive, it returns the following error message: "Operation not possible now. Please wait for system to stabilize before retrying."

- **Forcing Failover on the Secondary Node**.
  If you run the force failover command from the secondary node, the secondary node becomes primary and the primary node becomes secondary. A force failover can occur only if the secondary node's health is good and it is not configured to stay secondary.

  If the secondary node cannot become the primary node, or if secondary node was configured to stay secondary (using the STAYSECONDARY option), the node displays the following error message: "Operation not possible as my state is invalid. View the node for more information."

- **Forcing Failover When Nodes Are in Listen Mode**.
  When the two nodes of an HA pair are running different versions of the system software, the node running the higher version switches to the listen mode. In this mode, neither command propagation nor synchronization works.

  Before upgrading the system software on both nodes, you should test the new version on one of the nodes. To do this, you need to force a failover on the system that has already been upgraded. The upgraded system then takes over as the primary node, but neither command propagation or synchronization occurs. Also, all connections need to be re-established.

To force failover on a node by using the command line interface

At the command prompt, type:

force HA failover

To force failover on a node by using the configuration utility

Navigate to System > High Availability and, on the Nodes tab, select the node, in the Action list, select Force Failover.

Forcing the Secondary Node to Stay Secondary

Aug 28, 2013

In an HA setup, the secondary node can be forced to stay secondary regardless of the state of the primary node.

For example, suppose the primary node needs to be upgraded and the process will take a few seconds. During the upgrade, the primary node may go down for a few seconds, but you do not want the secondary node to take over; you want it to remain the secondary node even if it detects a failure in the primary node.

When you force the secondary node to stay secondary, it will remain secondary even if the primary node goes down. Also, when you force the status of a node in an HA pair to stay secondary, it does not participate in HA state machine transitions. The status of the node is displayed as STAYSECONDARY.

Forcing the node to stay secondary works on both standalone and secondary nodes. On a standalone node, you must use this option before you can add a node to create an HA pair. When you add the new node, the existing node continues to function as the primary node, and the new node becomes the secondary node.

Note: When you force a system to remain secondary, the forcing process is not propagated or synchronized. It affects only the node on which you run the command.

To force the secondary node to stay secondary by using the command line interface

At the command prompt, type:

set ha node -hastatus STAYSECONDARY

To force the secondary node to stay secondary by using the configuration utility

Navigate to System > High Availability, on the Nodes tab, open the local node, and select STAY SECONDARY.

Forcing the Primary Node to Stay Primary

Aug 28, 2013

In an HA setup, you can force the primary node to remain primary even after a failover. You can enable this option either on a primary node in an HA pair or on a standalone system.

On a standalone system, you must run this command before you can add a node to create an HA pair. When you add the new node, it becomes the primary node. The existing node stops processing traffic and becomes the secondary node in the HA pair.

To force the primary node to stay primary by using the command line interface

At the command prompt, type:

set ha node -hastatus STAYPRIMARY

To force the primary node to stay primary by using the configuration utility

Navigate to System > High Availability, on the Nodes tab, open the local node, and select STAY PRIMARY.

Understanding the High Availability Health Check Computation

Mar 19, 2012

The following table summarizes the factors examined in a health check computation:

- State of the CIs
- State of the FISs
- State of the route monitors

The following table summarizes the health check computation.

Table 1. High Availability Health Check Computation

FIS	CI	Route monitor	Condition
N	Y	N	If the system has any CIs, all of those CIs must be UP.
Y	Y	N	If the system has any FISs, all of those FISs must be UP.
Y	Y	Y	If the system has any route monitors configured, all monitored routes must be present in the FIS.

High Availability FAQs

Jul 16, 2013

What are the various ports used to exchange the HA-related information between the nodes in an HA configuration?

In an HA configuration, both nodes use the following ports to exchange HA related information:

- UDP Port 3003, to exchange heartbeat packets.
- Port 3010, for synchronization and command propagation.

What are the conditions that trigger synchronization?

Synchronization is triggered by any of the following conditions:

- The incarnation number of the primary node, received by the secondary, does not match that of the secondary node. Note: Both nodes in an HA configuration maintain a counter called *incarnation number*, which counts the number of configurations in the node's configuration file. Each node sends its incarnation number to each other node in the heartbeat messages. The incarnation number is not incremented for the following commands:
  1. All HA configuration related commands. For example, add ha node, set ha node, and bind ha node.
  2. All Interface related commands. For example, set interface and unset interface.
  3. All channel-related commands. For example, add channel, set channel, and bind channel.
- The secondary node comes up after a restart.
- The primary node becomes secondary after a failover.

What configurations are not synced or propagated in an HA configuration in INC or non-INC mode?

The following commands are neither propagated nor synced to the secondary node:

- All node specific HA configuration commands. For example, add ha node, set ha node, and bind ha node.
- All Interface related configuration commands. For example, set interface and unset interface.
- All channel related configuration commands. For example, add channel, set channel, and bind channel.

What configurations are not synced nor propagated in an HA configuration in INC mode?

The following configurations are not synced or propagated. Each node has its own.

- MIPs
- SNIPs
- VLANs
- Routes (except LLB routes)
- Route monitors
- RNAT rules (except any RNAT rule with VIP as the NAT IP)
- Dynamic routing configurations.

Does a configuration added to the secondary node get synchronized on the primary?

No, a configuration added to the secondary node is not synchronized to the primary.

What could be the reason for both nodes claiming to be the primary in an HA configuration?

The most likely reason is that the primary and secondary nodes are both healthy but the secondary does not receive the heartbeat packets from the primary. The problem could be with the network between the nodes.

Does an HA configuration run into any issues if you deploy the two nodes with different system clock settings?

Different system-clock settings on the two nodes can cause the following issues:

- The time stamps in the log file entries do not match. This situation makes it difficult to analyze the log entries for any issues.
- After a failover, you might have problems with any type of cookie based persistence for load balancing. A significant difference between the times can cause a cookie to expire sooner than expected, resulting in termination of the persistence session.
- Similar considerations apply to any time related decisions on the nodes.

What are the conditions for failure of the *force HA sync* command?

Forced synchronization fails in any of the following circumstances:

- You force synchronization when synchronization is already in progress.
- You force synchronization on a standalone NetScaler appliance.
- The secondary node is disabled.
- HA synchronization is disabled on the current secondary node.
- HA propagation is disabled on the current primary node and you force synchronization from the primary.

What are the conditions for failure of the *sync HA files* command?

Synchronizing configuration files fail in either of the following circumstances:

- On a standalone system.
- With the secondary node disabled.

In an HA configuration, if the secondary node takes over as the primary, does it switch back to secondary status if the original primary comes back online?

No. After the secondary node takes over as the primary, it remains as primary even if the original primary node comes back online again. To interchange the primary and secondary status of the nodes, run the *force failover* command.

What are the conditions for failure of the *force failover* command?

A forced failover fails in any of the following circumstances:

- You force failover on a standalone system.
- The secondary node is disabled.
- The secondary node is configured to remain secondary.
- The primary node is configured to remain primary.
- The state of the peer node is unknown.

Troubleshooting High Availability Issues

Jul 29, 2013

The most common high availability issues involve the high availability feature not working at all, or working only intermittently. Following are common high availability issues, and probable causes and resolutions.

- **Issue**
  The inability of the NetScaler appliances to pair the NetScaler appliances in a high availability setup.
  - **Cause**
    Network connectivity

    Resolution

    Verify that both the appliances are connected to the switch and the interfaces are enabled.

  - **Cause**
    Mismatch in the Password for the default Administrator account

    Resolution

    Verify that the password on both the appliances is the same.

  - **Cause**
    IP conflict

    Resolution

    Verify that both the appliances have unique NetScaler IP (NSIP) address. The appliances should not have the same NSIP address.

  - **Cause**
    Node ID mismatch

    Resolution

    Verify that the Node ID Configuration on both the appliances is unique. The appliances should not have the same Node ID configuration. Additionally, you must assign value for a Node ID between 1 and 64.

  - **Cause**
    Mismatch in the password of the RPC node

    Resolution

    Verify that both the nodes have the same RPC node password.

  - **Cause**
    An administrator has disabled the remote node

    Resolution

    Enable the remote node.

- **Cause**
  The Firewall application has blocked the heartbeat packets

  Resolution

  Verify that the UDP port 3003 is allowed.

- **Issue**
  Both the appliances claim to be the primary appliance.
  - **Cause**
    Missing heartbeat packets between the appliances

    Resolution

    Verify that the UDP port 3003 is not blocked for communication between the appliances.

- **Issue**
  The NetScaler appliance is not able to synchronize the configuration.
  - **Cause**
    A Firewall application is blocking the required port.

    Resolution

    Verify that the UDP port 3010 (or UDP port 3008 with secure synchronization) is not blocked for communication between the appliances.

  - **Cause**
    An administrator has disabled synchronization.

    Resolution

    Enable synchronization on the appliance that has the issue.

  - **Cause**
    Different NetScaler releases or builds are installed on appliances.

    Resolution

    Upgrade the appliances to the same NetScaler release or build.

- **Issue**
  Command propagation fails between the appliances.
  - **Cause**
    A Firewall application is blocking the port.

    Resolution

    Verify that the UDP port 3011 (or UDP port 3009 with secure propagation) is not blocked for communication between the appliances.

  - **Cause**
    An administrator has disabled command propagation.

Resolution

Enable command propagation on the appliance that has the issue.

- **Cause**
  Different NetScaler releases or builds are installed on appliances.

  Resolution

  Upgrade the appliances to the same NetScaler release or build.

- **Issue**
  The NetScaler appliances in the high availability pair are unable to run the force failover process.
  - **Cause**
    The Secondary node is disabled.

    Resolution

    Enable the secondary node.

  - **Cause**
    The Secondary node is configured to stay secondary.

    Resolution

    Set the secondary high availability status of the secondary node to Enable from Stay Secondary.

- **Issue**
  The secondary appliance does not receive any traffic after the failover process.
  - **Cause**
    The upstream router does not understand GARP messages of NetScaler appliance.

    Resolution

    Configure Virtual MAC (VMAC) address on the secondary appliance.

Networking

Sep 30, 2013

The following topics provide a conceptual reference and instructions for configuring the various networking components on the NetScaler appliance.

IP Addressing	Learn the various types of NetScaler-owned IP addresses and how to create, customize, and remove them.
Interfaces	Configure some of the basic network configurations that must be done to get started.
Access Control Lists (ACLs)	Configure the different types of Access Control Lists and how to create, customize, and remove them.
IP Routing	Learn and configure the routing functionality of the NetScaler appliance, both static and dynamic.
Internet Protocol version 6 (IPv6)	Learn how the NetScaler appliance supports IPv6.
Traffic Domains	Learn and configure traffic domains to segment network traffic for different applications.

IP Addressing

Aug 28, 2013

Before you can configure the NetScaler appliance, you must assign the NetScaler IP Address (NSIP), also known as the Management IP address. You can also create other NetScaler-owned IP addresses for abstracting servers and establishing connections with the servers. In this type of configuration, the appliance serves as a proxy for the abstracted servers. You can also proxy connections by using network address translations (INAT and RNAT). When proxying connections, the appliance can behave either as a bridging (Layer 2) device or as a packet forwarding (Layer 3) device. To make packet forwarding more efficient, you can configure static ARP entries. For IPv6, you can configure neighbor discovery (ND).

This document includes the following information:

Configuring NetScaler-Owned IP Addresses

Mar 20, 2012

The NetScaler-owned IP Addresses—NetScaler IP Address (NSIP), Virtual IP Addresses (VIPs), Subnet IP Addresses (SNIPs), Mapped IP Addresses (MIPs), and Global Server Load Balancing Site IP Addresses (GSLBIPs)—exist only on the NetScaler appliance. The NSIP uniquely identifies the NetScaler on your network, and it provides access to the appliance. A VIP is a public IP address to which a client sends requests. The NetScaler terminates the client connection at the VIP and initiates a connection with a server. This new connection uses a SNIP or a MIP as the source IP address for packets forwarded to the server. If you have multiple data centers that are geographically distributed, each data center can be identified by a unique GSLBIP.

You can configure some NetScaler-owned IP addresses to provide access for management applications.

This document includes the following information:

- Configuring the NetScaler IP Address (NSIP)
- Configuring and Managing Virtual IP (VIP) Addresses
- Configuring ARP response Suppression for Virtual IP addresses (VIPs)
- Configuring Subnet IP Addresses (SNIPs)
- Configuring Mapped IP Addresses (MIPs)
- Configuring GSLB Site IP Addresses (GSLBIP)
- Removing a NetScaler-Owned IP Address
- Configuring Application Access Controls

Configuring the NetScaler IP Address (NSIP)

Aug 28, 2013

The NetScaler IP (NSIP) address is the IP address at which you access the NetScaler for management purposes. The NetScaler can have only one NSIP, which is also called the Management IP address. You must add this IP address when you configure the NetScaler for the first time. If you modify this address, you must reboot the NetScaler. You cannot remove an NSIP address. For security reasons, NSIP should be a non-routable IP address on your organization's LAN.

Note: Configuring the NetScaler IP address is mandatory.

To create the NetScaler IP address by using the command line interface

At the command prompt, type:

- set ns config [-IPAddress <ip_addr> -netmask <netmask>]
- show ns config

Example

```
> set ns config -ipaddress 10.102.29.170 -netmask 255.255.255.0
 Done
```

To configure the NetScaler IP address by using the configuration utility

1. In the navigation pane, click System.
2. On the System Information tab, click Setup Wizard.
3. In the Setup Wizard dialog box, click Next.
4. Under System Configuration, set the following parameters:
   - IP Address
   - Netmask
5. Follow the instructions in the Setup Wizard to complete the configuration.

Configuring and Managing Virtual IP (VIP) Addresses

Nov 23, 2015

Configuration of a virtual server IP (VIP) address is not mandatory during initial configuration of the NetScaler ADC. When you configure load balancing, you assign VIP addresses to virtual servers.

For more information about configuring a load balancing setup, see "Load Balancing."

In some situations, you need to customize VIP attributes or enable or disable a VIP address. A VIP address is usually associated with a virtual server, and some of the VIP attributes are customized to meet the requirements of the virtual server. You can host the same virtual server on multiple NetScaler appliances residing on the same broadcast domain, by using ARP and ICMP attributes. After you add a VIP (or any IP address), the NetScaler sends, and then responds to, ARP requests. VIPs are the only NetScaler-owned IP addresses that can be disabled. When a VIP address is disabled, the virtual server using it goes down and does not respond to ARP, ICMP, or L4 service requests.

As an alternative to creating VIP addresses one at a time, you can specify a consecutive range of VIP addresses.

To create a VIP address by using the command line interface

At the command prompt, type:

- add ns ip <IPAddress> <netmask> -type <type>
- show ns ip <IPAddress>

Example

```
> add ns ip 10.102.29.59 255.255.255.0 -type VIP
 Done
```

To create a range of VIP addresses by using the command line interface

At the command prompt, type:

- add ns ip <IPAddress> <netmask> -type <type>
- show ns ip <IPAddress>

Example

```
> add ns ip 10.102.29.[60-64] 255.255.255.0 -type VIP
ip "10.102.29.60" added
ip "10.102.29.61" added
ip "10.102.29.62" added
ip "10.102.29.63" added
ip "10.102.29.64" added
 Done
```

To configure a VIP address by using the configuration utility

Navigate to System > Network > IPs > IPV4s, and add a new IP address or edit an existing address.

To create a range of VIP addresses by using the configuration utility

1. Navigate to System > Network > IPs > IPV4s.
2. In the Action list, select Add Range.

To enable or disable an IPv4 VIP address by using the command line interface

At the command prompt, type one of the following sets of commands to enable or disable a VIP and verify the configuration:

- enable ns ip <IPAddress>
- show ns ip <IPAddress>
- disable ns ip <IPAddress>
- show ns ip <IPAddress>

Example

```
> enable ns ip 10.102.29.79
 Done
> show ns ip 10.102.29.79

      IP: 10.102.29.79
      Netmask: 255.255.255.255
      Type: VIP
      state: Enabled
      arp: Enabled
      icmp: Enabled
      vserver: Enabled
      management access: Disabled
        telnet: Disabled
        ftp: Disabled
        ssh: Disabled
        gui: Disabled
        snmp: Disabled
      Restrict access: Disabled
      dynamic routing: Disabled
      hostroute: Disabled
 Done
> disable ns ip 10.102.29.79
 Done
> show ns ip 10.102.29.79

      IP: 10.102.29.79
      Netmask: 255.255.255.255
      Type: VIP
      state: Disabled
      arp: Enabled
      icmp: Enabled
      vserver: Enabled
      management access: Disabled
        telnet: Disabled
```



```
    ftp: Disabled
    ssh: Disabled
    gui: Disabled
    snmp: Disabled
  Restrict access: Disabled
  dynamic routing: Disabled
  hostroute: Disabled

 Done
```

To enable or disable a VIP address by using the configuration utility

1. Navigate to **System** > **Network** > **IPs** > **IPV4s**.
2. Do one of the following:
   - Select a VIP address.
   - Hold down the **Ctrl** key and select multiple address entries.
   - Hold down the **Shift** key and select a range of address entries.
   - Select all the address entries by selecting the check box on the left side of the header row.
3. Fom the **Action** list, select **Disable** or **Enable**.

Configuring ARP response Suppression for Virtual IP addresses (VIPs)

Aug 28, 2013

You can configure the NetScaler appliance to respond or not respond to ARP requests for a Virtual IP (VIP) address on the basis of the state of the virtual servers associated with that VIP.

For example, if virtual servers V1, of type HTTP, and V2, of type HTTPs, share VIP address 10.102.29.45 on a NetScaler appliance, you can configure the appliance to not respond to any ARP request for VIP 10.102.29.45 if both V1 and V2 are in the DOWN state.

The following three options are available for configuring ARP-response suppression for a virtual IP address.

- **NONE**. The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the state of the virtual servers associated with the address.
- **ONE VSERVER**. The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.
- **ALL VSERVER**. The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

Following table shows the sample behavior of NetScaler appliance for a VIP configured with two virtual servers:

Associated virtual servers for a VIP	**STATE 1**	**STATE 2**	**STATE 3**	**STATE 4**
NONE				
V1	UP	UP	DOWN	DOWN
V2	UP	DOWN	UP	DOWN
Respond to an ARP request for this VIP?	Yes	Yes	Yes	Yes
ONE VSERVER				
V1	UP	UP	DOWN	DOWN
V2	UP	DOWN	UP	DOWN
Respond to an ARP request for this VIP?	Yes	Yes	Yes	No
ALL VSERVER				
V1	UP	UP	DOWN	DOWN

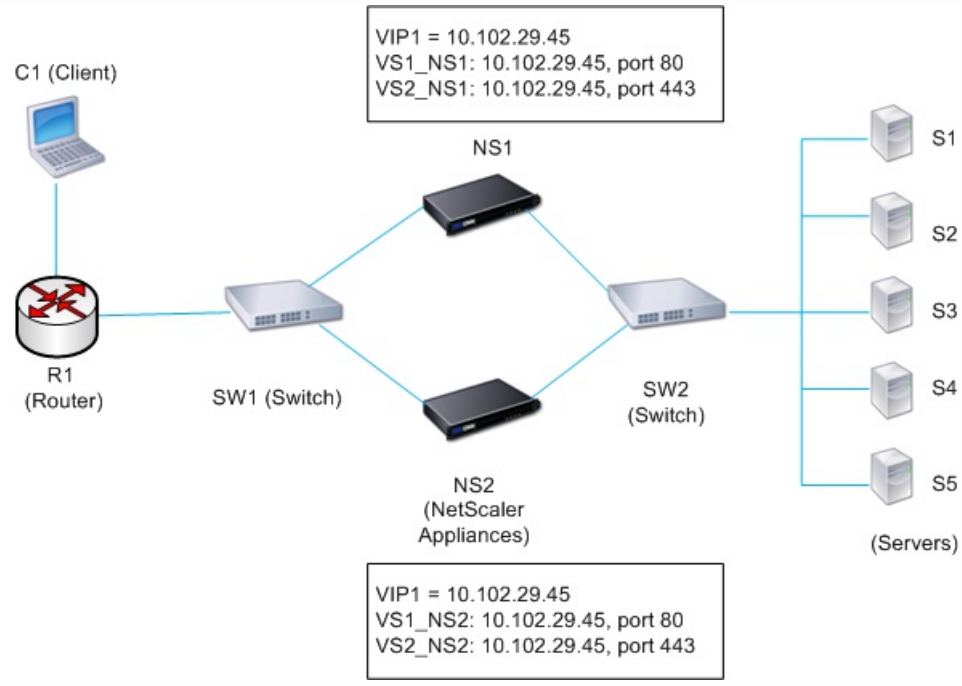
Associated virtual servers for a VIP	STATE 1	STATE 2	STATE 3	STATE 4
Respond to an ARP request for this VIP?	Yes	No	No	No

Consider an example where you want to test the performance of two virtual servers, V1 and V2, which have the same VIP address but are of different types and are each configured on NetScaler appliances NS1 and NS2. Let's call the shared VIP address *VIP1*.

V1 load balances servers S1, S2, and S3. V2 load balances servers S4 and S5.

On both NS1 and NS2, for VIP1, the ARP suppression parameter is set to ALL_VSERVER. If you want to test the performance of V1 and V2 on NS1, you must manually disable V1 and V2 on NS2, so that NS2 does not respond to any ARP request for VIP1.

Figure 1.



The execution flow is as follows:

1. Client C1 sends a request to V1. The request reaches R1.
2. R1 does not have an APR entry for the IP address (VIP1) of V1, so R1 broadcasts an ARP request for VIP1.
3. NS1 replies with source MAC address MAC1 and source IP address VIP1. NS2 does not reply to the ARP request.
4. SW1 learns the port for VIP1 from the ARP reply and updates its bridge table, and R1 updates the ARP entry with MAC1 and VIP1.
5. R1 forwards the packet to address VIP1 on NS1.
6. NS1's load balancing algorithm selects server S2, and NS1 opens a connection between one of its SNIP or MIP addresses and S2. When S2 sends a response to the client, the response returns by the same path.
7. Now you want to test the performance of V1 and V2 on NS2, so you enable V1 and V2 on NS2 and disable them on NS1. NS2 now broadcasts an ARP message for VIP1. In the message, MAC2 is the source MAC address and VIP1 is the source IP address.

8. SW1 learns the port number for reaching MAC2 from the ARP broadcast and updates its bridge table to send subsequent client requests for VIP1 to NS2. R1 updates its ARP table.
9. Now suppose the ARP entry for VIP1 times out in the ARP table of R1, and client C1 sends a request for V1. Because R1 does not have an APR entry for VIP1, it broadcasts an ARP request for VIP1.
10. NS2 replies with a source MAC address and VIP1 as the source IP address. NS1 does not reply to the ARP request.

To configure ARP response suppression by using the command line interface

At the command prompt, type:

- set ns ip -arpResponse <arpResponse>]
- show ns ip <IPAddress>

Example

```
> set ns ip 10.102.29.96 -arpResponse ALL_VSERVERS
 Done
```

To configure ARP response suppression by using the configuration utility

1. Navigate to System > Network > IPs > IPV4s.
2. Open an IP address entry and select the type of ARP Response.

Configuring Subnet IP Addresses (SNIPs)

May 23, 2014

A subnet IP address (SNIP) is a NetScaler owned IP address that is used by the NetScaler ADC to communicate with the servers.

The NetScaler ADC uses the subnet IP address as a source IP address to proxy client connections to servers. It also uses the subnet IP address when generating its own packets, such as packets related to dynamic routing protocols, or to send monitor probes to check the health of the servers.

Depending on your network topology, you might have to configure one or more SNIPs for different scenarios. Following are three typical scenarios in which you have to configure SNIPs:

- Using SNIPs for a Directly Connected Server Subnet
- Using SNIPs for Server Subnets Connected through a Router
- Using SNIPs for Multiple Server Subnets (VLANs) on an L2 Switch

To configure a SNIP address on a NetScaler ADC, you add the SNIP address and then enable global Use Subnet IP (USNIP) mode.

As an alternative to creating SNIPs one at a time, you can specify a consecutive range of SNIPs.

To configure a SNIP address by using the command line interface

At the command prompt, type:

- add ns ip <IPAddress> <netmask> -type SNIP
- show ns ip <IPAddress>

Example

```
> add ns ip 10.102.29.203 255.255.255.0 -type SNIP
 Done
```

To create a range of SNIP addresses by using the command line interface

At the command prompt, type:

- add ns ip <IPAddress> <netmask> -type SNIP
- show ns ip <IPAddress>

Example

```
> add ns ip 10.102.29.[205-209] 255.255.255.0 -type SNIP
ip "10.102.29.205" added
ip "10.102.29.206" added
ip "10.102.29.207" added
ip "10.102.29.208" added
ip "10.102.29.209" added
 Done
```

To enable or disable USNIP mode by using the command line interface

At the command prompt, type one of the following commands:

- enable ns modeUSNIP
- disable ns modeUSNIP

To configure a SNIP address by using the configuration utility

Navigate to System > Network > IPs > IPV4s, and add a new SNIP address or edit an existing address.

To create a range of SNIP addresses by using the configuration utility

1. Navigate to System > Network > IPs > IPV4s.
2. In the Action list, select Add Range.

To enable or disable USNIP mode by using the command line interface

At the command prompt, type one of the following commands:

- enable ns mode USNIP
- disable ns mode USNIP

To enable or disable USNIP mode by using the configuration utility

1. Navigate to System > Settings, in Modes and Features group, click Change modes.
2. Select or clear the Use Subnet IP option.

Using SNIPs for a Directly Connected Server Subnet

Updated: 2014-05-23

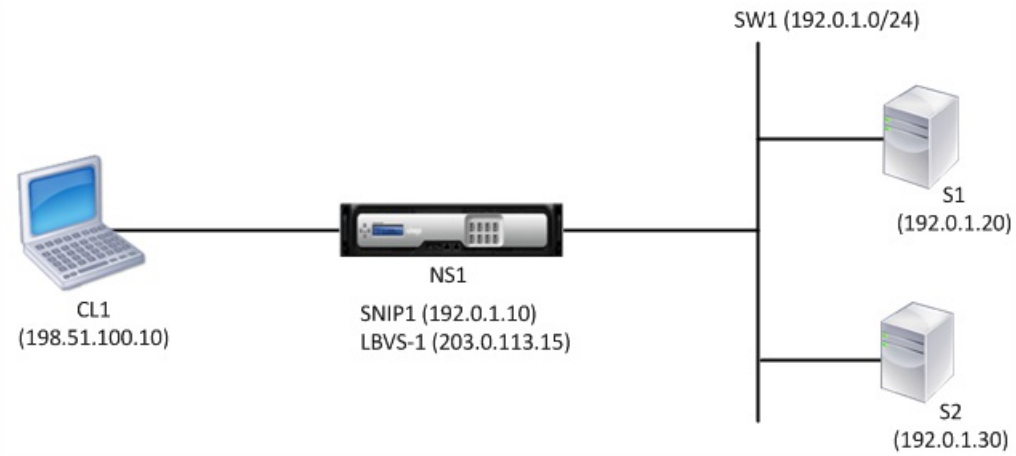
To enable communication between the NetScaler and a server that is either connected directly to the NetScaler or connected through only an L2 switch, you must configure a subnet IP address that belongs to the subnet of the server. You must configure at least one subnet IP address for each directly connected subnet, except for the directly connected management subnet that is connected through NSIP.

Consider an example of a load balancing setup in which load balancing virtual server LBVS1 on NetScaler ADC NS1 is used to load balance servers S1 and S2, which are connected to NS1 through L2 switch SW1. S1 and S2 belong to the same subnet.

SNIP address SNIP1, which belongs to the same subnet as S1 and S2, is configured on NS1. As soon as SNIP1 is configured, NS1 broadcasts ARP packets for SNIP1.

Services SVC-S1 and SVC-S2 on NS1 represent S1 and S2. As soon as these services are configured, NS1 broadcasts ARP requests for S1 and S2 to resolve IP-to-MAC mapping. After S1 and S2 respond, NS1 sends them monitoring probes at regular intervals, from address SNIP1, to check their health.

For more information about configuring load balancing on a NetScaler ADC, see Load Balancing.



Following is the traffic flow in this example:

1. Client C1 sends a request packet to LBVS-1. The request packet has:
   - Source IP = IP address of the client (198.51.100.10)
   - Destination IP = IP address of LBVS-1 (203.0.113.15)
2. LBVS1 of NS1 receives the request packet.
3. LBVS1's load balancing algorithm selects server S2.
4. Because S2 is directly connected to NS1, and SNIP1 (192.0.1.10) is the only IP address on NS1 that belongs to the same subnet as S2, NS1 opens a connection between SNIP1 and S2.
5. NS1 sends the request packet to S2 from SNIP1. The request packet has:
   - Source IP = SNIP1 (192.0.1.10)
   - Destination IP = IP address of S2 (192.0.1.30)
6. S2's response returns by the same path.

Using SNIPs for Server Subnets Connected through a Router

Updated: 2014-05-23

To enable communication between the NetScaler ADC and servers in subnets connected through a router, you must configure at least one subnet IP address that belongs to the subnet of the directly connected interface to the router. The ADC uses this subnet IP address to communicate with servers in subnets that can be reached through the router.

Consider an example of a load balancing setup in which load balancing virtual server LBVS1 on NetScaler ADC NS1 is used to load balance servers S1, S2, S3, and S4, which are connected to NS1 through router R1.

S1 and S2 belong to same subnet, 192.0.2.0/24, and are connected to R1 through L2 switch SW1. S3 and S4 belong to a different subnet, 192.0.3.0/24, and are connected to R1 through L2 switch SW2.

NetScaler ADC NS1 is connected to router R1 through subnet 192.0.1.0/24. SNIP address SNIP1, which belongs to the same subnet as the directly connected interface to the router (192.0.1.0/24), is configured on NS1. NS1 uses this address to communicate with servers S1 and S2, and with servers S3 and S4.

For more information about configuring load balancing on a NetScaler ADC, see Load Balancing.

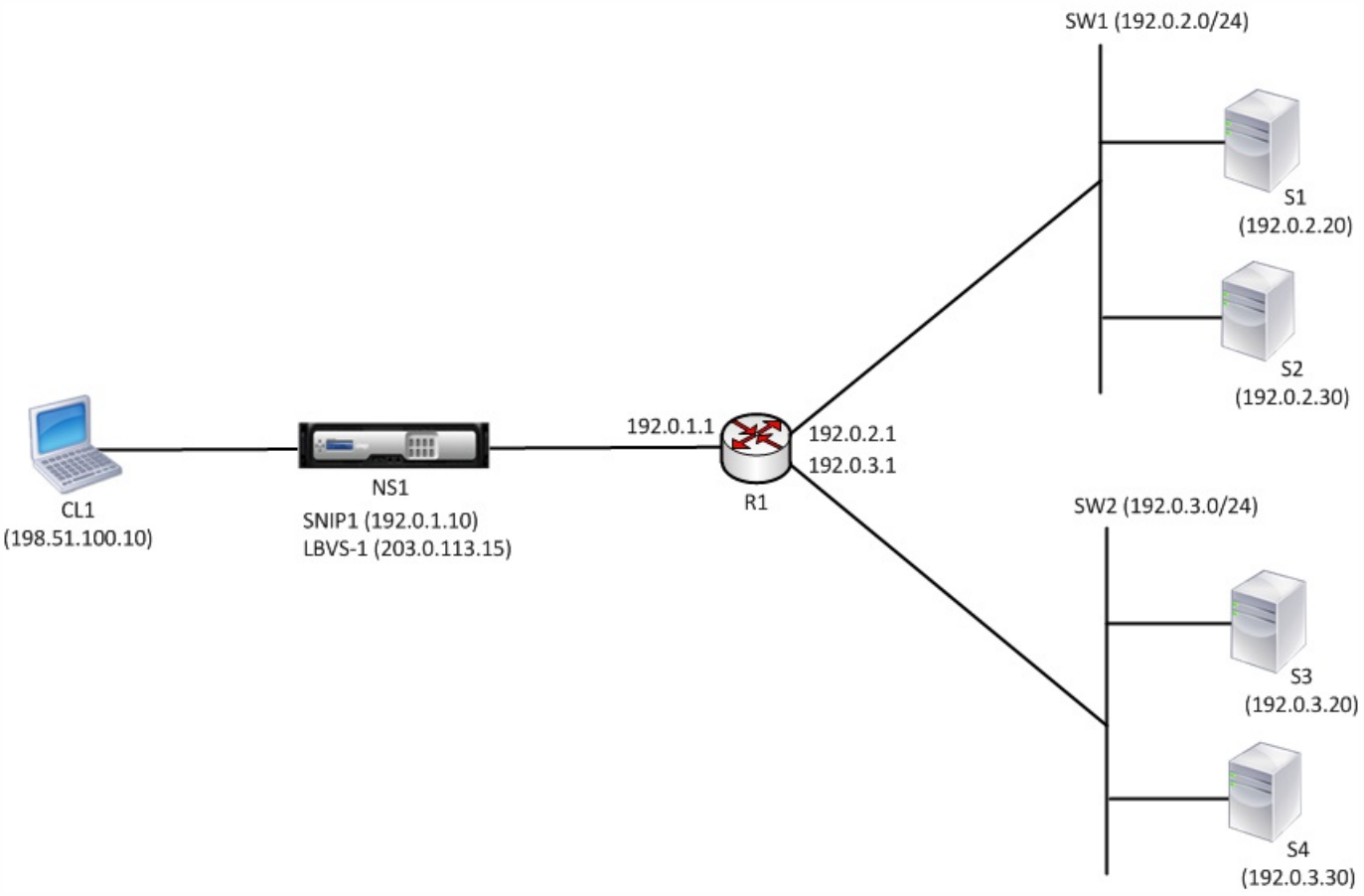
As soon as address SNIP1 is configured, NS1 broadcasts ARP announcement packets for SNIP1.

NS1's routing table consists of route entries for S1, S2, S3, and S4 through R1. These route entries are either static route entries or advertised by R1 to NS1, using dynamic routing protocols.

Services SVC-S1, SVC-S2, SVC-S3, and SVC-S4 on NS1 represent servers S1, S2, S3, and S4. NS1 finds, in its routing tables,

that these servers are reachable through R1. NS1 sends them monitoring probes at regular intervals, from address SNIP1, to check their health.

For more information about IP routing on a NetScaler ADC, see IP Routing.



Following is the traffic flow in this example:

1. Client C1 sends a request packet to LBVS-1. The request packet has:
   - Source IP = IP address of the client (198.51.100.10)
   - Destination IP = IP address of LBVS-1 (203.0.113.15)
2. LBVS1 of NS1 receives the request packet.
3. LBVS1's load balancing algorithm selects server S3.
4. NS1 checks its routing table and finds that S3 is reachable through R1. SNIP1 (192.0.1.10) is the only IP address on NS1 that belongs to the same subnet as router R1, NS1 opens a connection between SNIP1 and S3 through R1.
5. NS1 sends the request packet to R1 from SNIP1. The request packet has:
   - Source IP address = SNIP1 (192.0.1.10)
   - Destination IP address = IP address of S3 (192.0.3.20)
6. The request reaches R1, which checks its routing table and forwards the request packet to S3.
7. S3's response returns by the same path.

Using SNIPs for Multiple Server Subnets (VLANs) on an L2 Switch

Updated: 2014-05-23

When you have multiple server subnets (VLANs) on an L2 switch that is connected to a NetScaler ADC, you must configure at least one SNIP address for each of the server subnets, so that the NetScaler ADC can communicate with these server subnets.

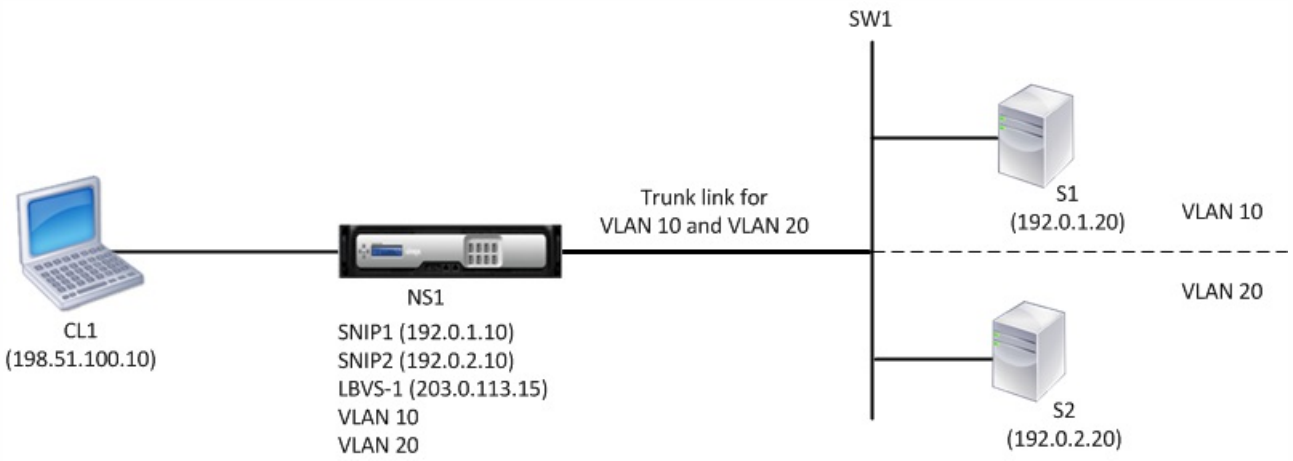
Consider an example of a load balancing setup in which load balancing virtual server LBVS1 on NetScaler ADC NS1 is used to load balance servers S1 and S2, which are connected to NS1 through L2 switch SW1. S1 and S2 belong to different subnets and are part of VLAN 10 and VLAN20, respectively. The link between NS1 and SW1 is a trunk link and is shared by VLAN10 and VLAN20.

For more information about configuring load balancing on a NetScaler ADC, see Load Balancing.

Subnet IP addresses SNIP1 (for reference purposes only) and SNIP2 (for reference purposes only) are configured on NS1. NS1 uses SNIP1 (on VLAN 10) to communicate with server S1, and SNIP2 (on VLAN 20) to communicate with S2. As soon as SNIP1 and SNIP2 are configured, NS1 broadcasts ARP announcement packets for SNIP1 and SNIP2.

For more information about configuring VLANs on a NetScaler ADC, see Configuring a VLAN.

Services SVC-S1 and SVC-S2 on NS1 represent servers S1 and S2. As soon as these services are configured, NS1 broadcasts ARP requests for them. After S1 and S2 respond, NS1 sends them monitoring probes at regular intervals to check their health. NS1 sends monitoring probes to S1 from address SNIP1, and to S2 from address SNIP2.



Following is the traffic flow in this example:

1. Client C1 sends a request packet to LBVS-1. The request packet has:
   - Source IP = IP address of the client (198.51.100.10)
   - Destination IP = IP address of LBVS-1 (203.0.113.15)
2. LBVS1 of NS1 receives the request packet.
3. LBVS1's load balancing algorithm selects server S2.
4. Because S2 is directly connected to NS1, and SNIP2 (192.0.2.10) is the only IP address on NS1 that belongs to the same subnet as S2, NS1 opens a connection between SNIP2 and S2.
   Note: If S1 is selected, NS1 opens a connection between SNIP1 and S1.
5. NS1 sends the request packet to S2 from SNIP2. The request packet has:
   - Source IP = SNIP1 (192.0.2.10)
   - Destination IP = IP address of S2 (192.0.2.20)
6. S2's response returns by the same path.

Configuring Mapped IP Addresses (MIPs)

Sep 30, 2013

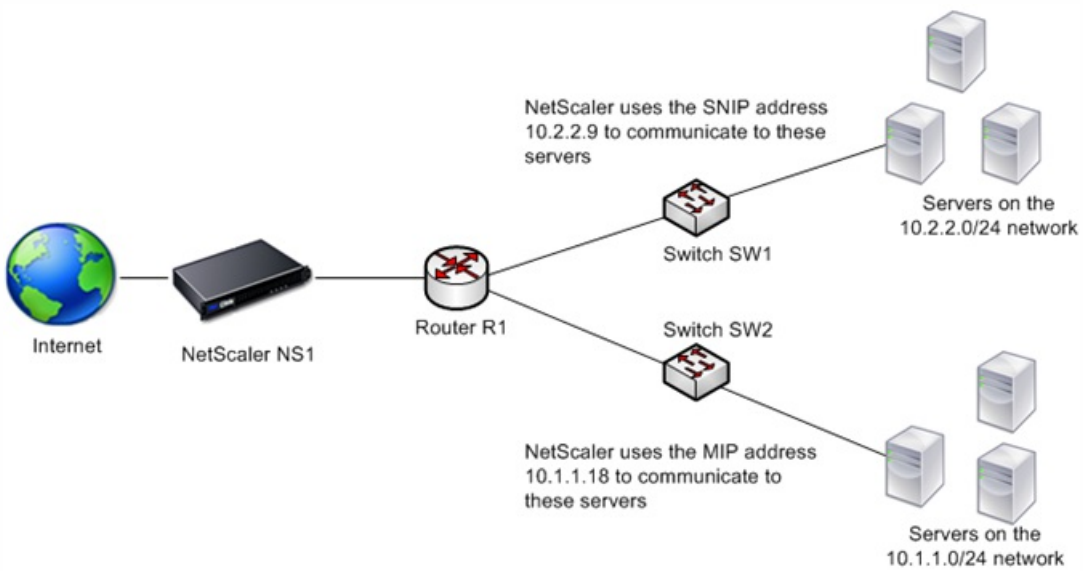
Mapped IP addresses (MIP) are used for server-side connections. A MIP can be considered a default Subnet IP (SNIP) address, because MIPs are used when a SNIP is not available or Use SNIP (USNIP) mode is disabled.

If the mapped IP address is the first in the subnet, the NetScaler appliance adds a route entry, with this IP address as the gateway to reach the subnet. You can create or delete a MIP during run time without rebooting the appliance.

As an alternative to creating MIPs one at a time, you can specify a consecutive range of MIPs.

The following diagram shows the use of the MIP and SNIP addresses in a NetScaler appliance that connects to the backend servers across the subnets.

Figure 1. MIP and SNIP addresses



In the setup, if the NetScaler appliance and the backend servers are in the 10.1.1.0/24 subnet, then the appliance uses the MIP address to communicate to the servers. However, if the setup has backend servers on additional subnets, such as 10.2.2.0/24, and there is no router between the NetScaler appliance and the subnet, then you can configure a SNIP address that has a range of 10.2.2.x/24, such as 10.2.2.9 in this case, to communicate to the additional subnet.

You can enable to NetScaler appliance to use MIP to communicate the additional subnet. However, if the setup has a Firewall application between the appliance and the server, then the Firewall might prevent the traffic other than 10.2.2.0/24. In such cases, you need a SNIP address to communicate to the servers.

To create a MIP address by using the command line interface

At the command prompt, type:

- add ns ip <IPAddress> <netmask> -type <type>
- show ns ip <IPAddress>

Example

```
> add ns ip 10.102.29.171 255.255.255.0 -type MIP
 Done
```

To create a range of MIP addresses by using the command line interface

At the command prompt, type:

- add ns ip <IPAddress> <netmask> -type <type>
- show ns ip <IPAddress>

Example

```
> add ns ip 10.102.29.[173-175] 255.255.255.0 -type MIP
ip "10.102.29.173" added
ip "10.102.29.174" added
ip "10.102.29.175" added
Done
```

To configure a MIP address by using the configuration utility

Navigate to System > Network > IPs > IPV4s, and add a new MIP address or edit an existing address.

To create a range of MIP addresses by using the configuration utility

1. Navigate to System > Network > IPs > IPV4s.
2. In the Action list, select Add Range.

Configuring GSLB Site IP Addresses (GSLBIP)

Aug 28, 2013

A GSLB site IP (GSLBIP) address is an IP address associated with a GSLB site. It is not mandatory to specify a GSLBIP address when you initially configure the NetScaler appliance. A GSLBIP address is used only when you create a GSLB site.

For more information about creating a GSLB site IP address, see "Global Server Load Balancing."

Removing a NetScaler-Owned IP Address

Aug 28, 2013

You can remove any IP address except the NSIP. The following table provides information about the processes you must follow to remove the various types of IP addresses. Before removing a VIP, remove the associated virtual server.

Table 1. Implications of Removing a NetScaler-Owned IP Address

IP address type	Implications
Subnet IP address (SNIP)	If IP address being removed is the last IP address in the subnet, the associated route is deleted from the route table. If the IP address being removed is the gateway in the corresponding route entry, the gateway for that subnet route is changed to another NetScaler-owned IP address.
Mapped IP address (MIP)	If a SNIP exists, you can remove the MIPs. The NetScaler uses NSIP and SNIPs to communicate with the servers when the MIP is removed. Therefore, you must also enable use SNIP (USNIP) mode. For information about enabling and disabling USNIP mode, see "Configuring Subnet IP Addresses (SNIPs)."
Virtual Server IP address (VIP)	Before removing a VIP, you must first remove the vserver associated with it. For information about removing the vserver, see "Load Balancing."
GSLB-Site-IP address	Before removing a GSLB site IP address, you must remove the site associated with it. For information about removing the site, see "Global Server Load Balancing."

To remove an IP address by using the command line interface

At the command prompt, type:

rm ns ip <IPaddress>

Example

```
rm ns ip 10.102.29.54
```

To remove an IP address by using the configuration utility

Navigate to System > Network > IPs > IPV4s, delete the IP address.

Configuring Application Access Controls

Sep 30, 2015

Application access controls, also known as management access controls, form a unified mechanism for managing user authentication and implementing rules that determine user access to applications and data. You can configure MIPs and SNIPs to provide access for management applications. Management access for the NSIP is enabled by default and cannot be disabled. You can, however, control it by using ACLs.

For information about using ACLs, see "Access Control Lists (ACLs)."

The NetScaler appliance does not support management access to VIPs.

The following table provides a summary of the interaction between management access and specific service settings for Telnet.

Management Access	Telnet (State Configured on the NetScaler)	Telnet (Effective State at the IP Level)
Enable	Enable	Enable
Enable	Disable	Disable
Disable	Enable	Disable
Disable	Disable	Disable

The following table provides an overview of the IP addresses used as source IP addresses in outbound traffic.

Application/ IP	NSIP	MIP	SNIP	VIP
ARP	Yes	Yes	Yes	No
Server side traffic	No	Yes	Yes	No
RNAT	No	Yes	Yes	Yes
ICMP PING	Yes	Yes	Yes	No
Dynamic routing	Yes	No	Yes	Yes

The following table provides an overview of the applications available on these IP addresses.

Application/ IP	NSIP	MIP	SNIP	VIP

Application/ IP	NSIP	MIP	SNIP	VIP
SNMP	Yes	Yes	Yes	Yes
System access	Yes	Yes	Yes	No

You can access and manage the NetScaler by using applications such as Telnet, SSH, GUI, and FTP.

Note: Telnet and FTP are disabled on the NetScaler for security reasons. To enable them, contact the customer support. After the applications are enabled, you can apply the controls at the IP level.
To configure the NetScaler to respond to these applications, you need to enable the specific management applications. If you disable management access for an IP address, existing connections that use the IP address are not terminated, but no new connections can be initiated.

Also, the non-management applications running on the underlying FreeBSD operating system are open to protocol attacks, and these applications do not take advantage of the NetScaler appliance's attack prevention capabilities.

You can block access to these non-management applications on a MIP, SNIP, or NSIP. When access is blocked, a user connecting to a NetScaler by using the MIP, SNIP, or NSIP is not be able to access the non-management applications running on the underlying operating system.

To configure management access for an IP address by using the command line interface

At the command prompt, type:

set ns ip <IPAddress> -mgmtAccess <value> -telnet <value> -ftp <value> -gui <value> -ssh <value> -snmp <value> -restrictAccess (ENABLED | DISABLED)
Example

```
> set ns ip 10.102.29.54 -mgmtAccess enabled -restrictAccess ENABLED
Done
```

To enable management access for an IP address by using the configuration utility

1. Navigate to System > Network > IPs > IPV4s.
2. Open an IP address entry, and select the Enable Management Access control to support the below listed applications option.

How the NetScaler Proxies Connections

Aug 28, 2013

When a client initiates a connection, the NetScaler appliance terminates the client connection, initiates a connection to an appropriate server, and sends the packet to the server. The appliance does not perform this action for service type UDP or ANY.

For more information about service types, see "Load Balancing."

You can configure the NetScaler to process the packet before initiating the connection with a server. The default behavior is to change the source and destination IP addresses of a packet before sending the packet to the server. You can configure the NetScaler to retain the source IP address of the packets by enabling Use Source IP mode.

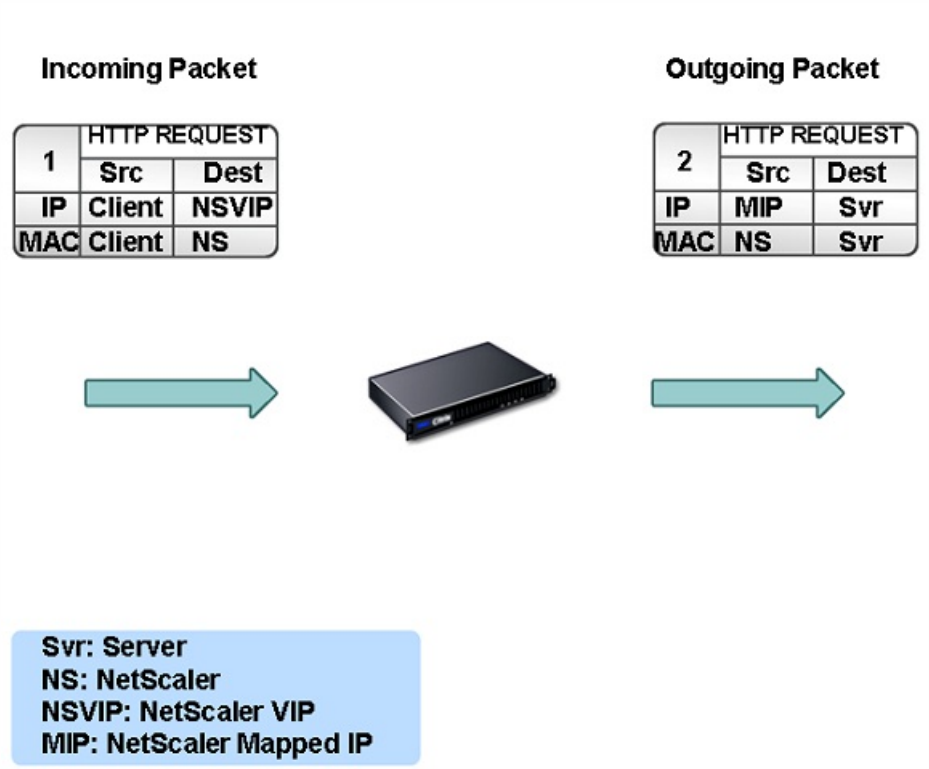
This section includes the following details:

- How the Destination IP Address Is Selected
- How the Source IP Address Is Selected

How the Destination IP Address Is Selected

Traffic sent to the NetScaler appliance can be sent to a virtual server or to a service. The appliance handles traffic to virtual servers and services differently. The NetScaler terminates traffic received at a virtual server IP (VIP) address and changes the destination IP address to the IP address of the server before forwarding the traffic to the server, as shown in the following diagram.

Figure 1. Proxying Connections to VIPs



Packets destined for a service are sent directly to the appropriate server, and the NetScaler does not modify the

destination IP addresses. In this case, the NetScaler functions as a proxy.

How the Source IP Address Is Selected

When the NetScaler appliance communicates with the physical servers or peer devices, by default, it does not use the IP address of the client. NetScaler maintains a pool of mapped IP addresses (MIPs) and subnet IP addresses (SNIPs), and selects an IP address from this pool to use as the source IP address of a connection to the physical server. Depending on the subnet in which the physical server is placed, NetScaler decides whether a MIP should be used or SNIP.

Note: If the Use Source IP (USIP) option is enabled, NetScaler uses the IP address of the client.

Enabling Use Source IP Mode

Dec 16, 2013

When the NetScaler appliance communicates with the physical servers or peer devices, by default, it uses one of its own IP addresses as the source IP. The appliance maintains a pool of mapped IP addresses (MIPs) and subnet IP addresses (SNIPs), and selects an IP address from this pool to use as the source IP address for a connection to the physical server. The decision of whether to select a MIP or a SNIP depends on the subnet in which the physical server resides.

If necessary, you can configure the NetScaler appliance to use the client's IP address as source IP. Some applications need the actual IP address of the client. The following use cases are a few examples:

- Client's IP address in the web access log is used for billing purposes or usage analysis.
- Client's IP address is used to determine the country of origin of the client or the originating ISP of the client. For example, many search engines such as Goggle provide content relevant to the location to which the user belongs.
- The application must know the client's IP address to verify that the request is from a trustworthy source.
- Sometimes, even though an application server does not need the client's IP address, a firewall placed between the application server and the NetScaler may need the client's IP address for filtering the traffic.

Enable Use Source IP mode (USIP) mode if you want NetScaler to use the client's IP address for communication with the servers. By default, USIP mode is disabled. USIP mode can be enabled globally on the NetScaler or on a specific service. If you enable it globally, USIP is enabled by default for all subsequently created services. If you enable USIP for a specific service, the client's IP address is used only for the traffic directed to that service.

As an alternative to USIP mode, you have the option of inserting the client's IP address (CIP) in the request header of the server-side connection for an application server that needs the client's IP address.

In earlier NetScaler releases, USIP mode had the following source-port options for server-side connections:

- Use the client's port. With this option, connections cannot be reused. For every request from the client, a new connection is made with the physical server.
- Use proxy port. With this option, connection reuse is possible for all requests from the same client. Before NetScaler release 8.1 this option imposed a limit of 64000 concurrent connections for all server-side connections.

In the later NetScaler releases , if USIP is enabled, the default is to use a proxy port for server-side connections and not reuse connections. Not reusing connections may not affect the speed of establishing connections.

By default, the Use Proxy Port option is enabled if the USIP mode is enabled.

For more information about the Use Proxy Port option, see "Using the Client Port When Connecting to the Server."

Note: If you enable the USIP mode, it is recommended to enable the Use Proxy Port option.
The following figure shows how the NetScaler uses IP addresses in USIP mode.

Figure 1. IP Addressing in USIP Mode

Recommended Usage

Enable USIP in the following situations:

- Load balancing of Intrusion Detection System (IDS) servers
- SMTP load balancing
- Stateless connection failover
- Sessionless load balancing
- If you use the Direct Server Return (DSR) mode

Note: When USIP is enabled, you must set server's gateway to one of the NetScaler owned IP addresses (either of type Subnet IP (SNIP) or mapped IP (MIP)) so that server's response always go through the NetScaler appliance. For more information about NetScaler owned IP addresses, see "Configuring NetScaler owned IP addresses."

- If you enable USIP, set the idle timeout for server connections to a value lower than the default value, so that idle connections are cleared quickly on the server side.
  For more information about setting an idle time-out value, "Load Balancing."
- For transparent cache redirection, if you enable USIP, enable L2CONN also.
- Because HTTP connections are not reused when USIP is enabled, a large number of server-side connections may accumulate. Idle server connections can block connections for other clients. Therefore, set limits on maximum number of connections to a service. Citrix also recommends setting the HTTP server time-out value, for a service on which USIP is enabled, to a value lower than the default, so that idle connections are cleared quickly on the server side.

To globally enable or disable USIP mode by using the command line interface

At the command prompt, type one of the following commands:

- enable ns mode USIP
- disable ns mode USIP

To enable USIP mode for a service by using the command line interface

At the command prompt, type:

set service <name>@ -usip (**YES** | **NO**)
Example

set service Service-HTTP-1 -usip YES

To globally enable or disable USIP mode by using the configuration utility

1. Navigate to System > Settings, in Modes and Features group, click Change modes.
2. Select or clear the Use Source IP option.

To enable USIP mode for a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to enable the USIP mode, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Settings, select the Use Source IP check box.
5. Click OK.

Configuring Network Address Translation

Jul 05, 2013

Network address translation (NAT) involves modification of the source and/or destination IP addresses and/or the TCP/UDP port numbers of IP packets that pass through the NetScaler appliance. Enabling NAT on the appliance enhances the security of your private network, and protects it from a public network such as the Internet, by modifying your networks source IP addresses when data passes through the NetScaler. Also, with the help of NAT entries, your entire private network can be represented by a few shared public IP addresses. The NetScaler supports the following types of network address translation:

- Inbound NAT (INAT), in which the NetScaler replaces the destination IP address in the packets generated by the client with the private IP address of the server.
- Reverse NAT (RNAT), in which the NetScaler replaces the source IP address in the packets generated by the servers with the public NAT IP addresses.

This document includes the following information:

Configuring INAT

Aug 28, 2013

When a client sends a packet to a NetScaler appliance that is configured for Inbound Network Address Translation (INAT), the appliance translates the packet's public destination IP address to a private destination IP address and forwards the packet to the server at that address.

The following configurations are supported:

- **IPv4-IPv4 Mapping**: A public IPv4 address on the NetScaler appliance listens to connection requests on behalf of a private IPv4 server. The NetScaler appliance translates the packet's public destination IP address to the destination IP address of the server and forwards the packet to the server at that address.
- **IPv4-IPv6 Mapping**: A public IPv4 address on the NetScaler appliance listens to connection requests on behalf of a private IPv6 server. The NetScaler appliance creates an IPv6 request packet with the IP address of the IPv6 server as the destination IP address.
- **IPv6-IPv4 Mapping**: A public IPv6 address on the NetScaler appliance listens to connection requests on behalf of a private IPv4 server. The NetScaler appliance creates an IPv4 request packet with the IP address of the IPv4 server as the destination IP address.
- **IPv6-IPv6 Mapping**: A public IPv6 address on the NetScaler appliance listens to connection requests on behalf of a private IPv6 server. The NetScaler appliance translates the packet's public destination IP address to the destination IP address of the server and forwards the packet to the server at that address.

When the appliance forwards a packet to a server, the source IP address assigned to the packet is determined as follows:

- If use subnet IP (USNIP) mode is enabled and use source IP (USIP) mode is disabled, the NetScaler uses a subnet IP address (SNIP) as the source IP address.
- If USNIP mode is disabled and USIP mode is disabled, the NetScaler uses a mapped IP address (MIP) as the source IP address.
- If USIP mode is enabled, and USNIP mode is disabled the NetScaler uses the client IP (CIP) address as the source IP address.
- If both USIP and USNIP modes are enabled, USIP mode takes precedence.
- You can also configure the NetScaler to use a unique IP address as the source IP address, by setting the proxyIP parameter.
- If none of the above modes are enabled and a unique IP address has not been specified, the NetScaler attempts to use a MIP as the source IP address.
- If both USIP and USNIP modes are enabled and a unique IP address has been specified, the order of precedence is as follows: USIP-unique IP-USNIP-MIP-Error.

To protect the NetScaler from DoS attacks, you can enable TCP proxy. However, if other protection mechanisms are used in your network, you may want to disable them.

You can create, modify, or remove an INAT entry.

To create an INAT entry by using the command line interface

At the command prompt, type the following commands to create an INAT entry and verify its configuration:

- add inat <name> <publicIP> <privateIP> [-tcpproxy (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-usip (ON | OFF)] [-usnip (ON | OFF)] [-proxyIP <ip_addr|ipv6_addr>]

- show inat [<name>]

Example

```
> add inat ip4-ip4 172.16.1.2 192.168.1.1 -proxyip 10.102.29.171
 Done
```

To modify an INAT entry by using the command line interface

To modify an INAT entry, type the set inat command, the name of the entry, and the parameters to be changed, with their new values.

To remove an INAT configuration by using the command line interface

At the command prompt, type:

rm inat <name>

Example

```
> rm inat ip4-ip4
 Done
```

To configure an INAT entry by using the configuration utility

Navigate to System > Network > Routes > INAT, and add a new INAT entry or edit an existing INAT entry.

To remove an INAT configuration by using the configuration utility

Navigate to System > Network > Routes > INAT, delete the INAT configuration.

Coexistence of INAT and Virtual Servers

Mar 20, 2012

If both INAT and RNAT are configured, the INAT rule takes precedence over the RNAT rule. If RNAT is configured with a network address translation IP (NAT IP) address, the NAT IP address is selected as the source IP address for that RNAT client.

The default public destination IP in an INAT configuration is the virtual IP (VIP) address of the NetScaler device. Virtual servers also use VIPs. When both INAT and a virtual server use the same IP address, the Vserver configuration overrides the INAT configuration.

Following are a few sample configuration setup scenarios and their effects.

Case	Result
You have configured a virtual server and a service to send all data packets received on a specific NetScaler port to the server directly. You have also configured INAT and enabled TCP. Configuring INAT in this manner sends all data packets received through a TCP engine before sending them to the server.	All packets received on the NetScaler, except those received on the specified port, pass through the TCP engine.
You have configured a virtual server and a service to send all data packets of service type TCP, that are received on a specific port on the NetScaler, to the server after passing through the TCP engine. You have also configured INAT and disabled TCP. Configuring INAT in this manner sends the data packets received directly to the server.	Only packets received on the specified port pass through the TCP engine.
You have configured a virtual server and a service to send all data packets received to either of two servers. You are attempting to configure INAT to send all data packets received to a different server.	The INAT configuration is not allowed.
You have configured INAT to send all received data packets directly to a server. You are attempting to configure a virtual server and a service to send all data packets received to two different servers.	The vserver configuration is not allowed.

Stateless NAT46 Translation

Aug 28, 2013

The stateless NAT46 feature enables communication between IPv4 and IPv6 networks through IPv4 to IPv6 packet translation, and vice versa, without maintaining any session information on the NetScaler appliance.

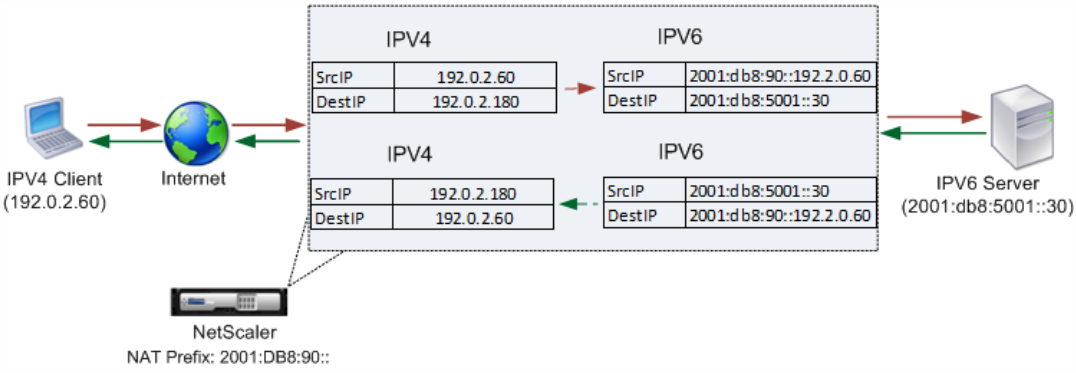
For a stateless NAT46 configuration, the appliance translates an IPv4 packet to IPv6 or an IPv6 packet to IPv4 as defined in RFCs 6145 and 2765.

Note: This feature is supported only on NetScaler 10.e and later.
A stateless NAT46 configuration on the NetScaler appliance has the following components:

- **IPv4-IPv6 INAT entry**—An INAT entry defining a 1:1 relationship between an IPv4 address and an IPv6 address. In other words, an IPv4 address on the appliance listens to connection requests on behalf of an IPv6 server. An IPv4 request packet for this IPv4 address is translated into an IPv6 packet, and then the IPv6 packet is sent to the IPv6 server.
  The appliance translates an IPv6 response packet into an IPv4 response packet with its source IP address field set as the IPv4 address specified in the INAT entry. The translated packet is then sent to the client.

- **NAT46 IPv6 prefix**—A global IPv6 prefix of length 96 bits (128-32=96) configured on the appliance. During IPv4 packet to IPv6 packet translation, the appliance sets the source IP address of the translated IPv6 packet to a concatenation of the NAT46 IPv6 prefix [96 bits] and the IPv4 source address [32 bits] that was received in the request packet.
  During IPv6 packet to IPv4 packet translation, the appliance sets the destination IP address of the translated IPv4 packet to the last 32 bits of the destination IP address of the IPv6 packet.

Consider an example in which an enterprise hosts site www.example.com on server S1, which has an IPv6 address. To enable communication between IPv4 clients and IPv6 server S1, NetScaler appliance NS1 is deployed with a stateless NAT46 configuration that includes an IPv4-IPv6 INAT entry for server S1, and a NAT46 Prefix. The INAT entry includes an IPv4 address at which the appliance listens to connection requests from IPv4 clients on behalf of the IPv6 server S1.



The following table lists the settings used in this example:

Entities	Name	Value
IP address of the client	Client_IPv4 (for reference purposes only)	192.0.2.60
IPv6 address of the server	Sevr_IPv6 (for reference purposes only)	2001:DB8:5001::30
IPv4 address defined in the INAT entry for IPv6 server S1	Map-Sevr-IPv4 (for reference purposes only)	192.0.2.180
IPv6 prefix for NAT 46 translation	NAT46_Prefix (for reference purposes only)	2001:DB8:90::

Following is the traffic flow in this example:

1. IPv4 Client CL1 sends a request packet to the Map-Sevr-IPv4 (192.0.2.180) address on the NetScaler appliance.
2. The appliance receives the request packet and searches the NAT46 INAT entries for the IPv6 address mapped to the Map-sevr-IPv4 (192.0.2.180) address. It finds the Sevr-IPv6 (2001:DB8:5001::30) address.
3. The appliance creates a translated IPv6 request packet with:
   - Destination IP address field = Sevr-IPv6 = 2001:DB8:5001::30
   - Source IP address field = Concatenation of NAT Prefix (First 96 bits) and Client_IPv4 (last 32 bits) = 2001:DB8:90::192.0.2.60
4. The appliance sends the translated IPv6 request to Sevr-IPv6.
5. The IPv6 server S1 responds by sending an IPv6 packet to the NetScaler appliance with:
   - Destination IP address field = Concatenation of NAT Prefix (First 96 bits) and Client_IPv4 (last 32 bits)= 2001:DB8:90::192.0.2.60
   - Source IP address field = Sevr-IPv6 = 2001:DB8:5001::30
6. The appliance receives the IPv6 response packet and verifies that its destination IP address matches the NAT46 prefix configured on the appliance. Because the destination address matches the NAT46 prefix, the appliance searches the NAT46 INAT entries for the IPv4 address associated with the Sevr-IPv6 address (2001:DB8:5001::30). It finds the Map-Sevr-IPv4 address (192.0.2.180).
7. The appliance creates an IPv4 response packet with:
   - Destination IP address field = The NAT46 prefix stripped from the destination address of the IPv6 response = Client_IPv4 (192.0.2.60)
   - Source IP address field = Map-Sevr-IPv4 address (192.0.2.180)
8. The appliance sends the translated IPv4 response to client CL1.

This section includes the following details:

- Configuring Stateless NAT46
- Setting Global Parameters for Stateless NAT46
- Limitations of Stateless NAT46

Configuring Stateless NAT46

Updated: 2013-09-04

Creating the required entities for stateless NAT46 configuration on the NetScaler appliance involves the following procedures:

1. Create an IPv4-IPv6 mapping INAT entry with stateless mode enabled.
2. Add a NAT46 IPv6 prefix.

To configure an INAT mapping entry by using the command line interface

At the command prompt, type:

- add inat <name> <publicIPv4> <privateIPv6> -mode STATELESS
- show inat <name>

To add an NAT46 prefix by using the command line interface

At the command prompt, type:

- set inatparam -nat46v6Prefix <ipv6_addr|*>
- show inatparam

Example

```
 > add inat exmpl-com-stls-nat46 192.0.2.180
2001:DB8:5001::30 -mode stateless
Done

> set inatparam -nat46v6Prefix 2001:DB8:90::/96
Done
```

To configure an INAT mapping entry by using the configuration utility

1. Navigate to System > Network > Routes > INAT.
2. Add a new INAT entry, or edit an existing INAT entry.
3. Set the following parameters:

- Name*
- Public IP Address*
- Private IP Address* (Select the IPv6 check box and enter the address in IPv6 format.)
- Mode (Select Stateless from the drop down list.)

* A required parameter

To add a NAT46 prefix by using the configuration utility

Navigate to System > Network, in the Settings group, click Configure INAT Parameters, and set the Prefix parameter.

Setting Global Parameters for Stateless NAT46

The appliance provides some optional global parameters for stateless NAT46 configurations.

To set global parameters for stateless NAT46 by using the command line interface

At the command prompt, type:

- set inatparam [-nat46IgnoreTOS (YES | NO)] [-nat46ZeroCheckSum (ENABLED | DISABLED)] [-nat46v6Mtu <positive_integer>] [-nat46FragHeader (ENABLED | DISABLED)]
- show inatparam

Example

```
> set inatparam -nat46IgnoreTOS YES -nat46ZeroCheckSum DISABLED -nat46v6Mtu 1400 -nat46FragHeader DISABLED
 Done
```

To set global parameters for stateless NAT46 by using the configuration utility

Navigate to System > Network, in the Settings group, click Configure INAT Parameters.

Limitations of Stateless NAT46

The following limitations apply to stateless NAT46:

- Translation of IPv4 options is not supported.
- Translation of IPv6 routing headers is not supported.
- Translation of hop-by-hop extension headers of IPv6 packets is not supported.
- Translation of ESP and EH headers of IPv4 packets is not supported.
- Translation of multicast packets is not supported.
- Translation of destination option headers and source routing headers is not supported.
- Translation of fragmented IPv4 UDP packets that do not contain UDP checksum is not supported.

DNS64

Feb 23, 2014

The NetScaler DNS64 feature responds with a synthesized DNS AAAA record to an IPv6 client sending an AAAA request for an IPv4-only domain. The DNS64 feature is used with the NAT64 feature to enable seamless communication between IPv6-only clients and IPv4-only servers. DNS64 enables discovery of the IPv4domain by the IPV6 only clients, and NAT64 enables communication between the clients and servers.

For synthesizing an AAAA record, the NetScaler appliance fetches a DNS A record from a DNS server. The DNS64 prefix is a 96-bit IPv6 prefix configured on the NetScaler appliance. The NetScaler appliance synthesizes the AAAA record by concatenation of the DNS64 Prefix (96 bits) and the IPv4 address (32 bits).

For enabling communication between IPv6 clients and IPv4 servers, a NetScaler appliance with DNS64 and NAT64 configuration can deployed either on the IPv6 client side or on the IPv4 server side. In both cases, the DNS64 configuration on the NetScaler appliance is similar and includes a load balancing virtual server acting as a proxy server for DNS servers. If the NetScaler appliance is deployed on the client side, the load balancing virtual server must be specified, on the IPv6 client, as the nameserver for a domain.

Consider an example where a NetScaler appliance with DNS64 and NAT64 configuration is configured on the IPv4 side. In this example, an enterprise hosts site www.example.com on server S1, which has an IPv4 address. To enable communication between IPv6 clients and IPv4 server S1, NetScaler appliance NS1 is deployed with a DNS64 and stateful NAT64 configuration.

The DNS64 configuration includes DNS load balancing virtual server LBVS-DNS64-1, on which the DNS64 option is enabled. A DNS64 policy named DNS64-Policy-1, and an associated DNS64 action named DNS64-Action-1, are also configured on NS1, andDNS64-Policy-1 is bound to LBVS-DNS64-1. LBVS-DNS64-1 acts as a DNS proxy server for DNS servers DNS-1 and DNS-2.

When traffic arriving at LBVS-DNS64-1 matches the conditions specified in DNS64-Policy-1, the traffic is processed according to the settings in DNS64-Action-1. DNS64-Action-1 specifies the DNS64 prefix used, with the A record received from a DNS server, to synthesize an AAAA record.

The global DNS parameter cacherecords is enabled on the NetScaler appliance, so the appliance caches DNS records. This setting is necessary for the DNS64 to work properly.

The following table lists the settings used in the above example:

Entity	Name	Value
IPv6 client	CL1 (for reference purposes only)	• IP address = 2001:DB8:5001::30
DNS64 Prefix		• 2001:DB8:300::
Service on NS representing DNS server DNS-1	SVC-DNS-1	• IP address = 203.0.113.50 • Port = 53

Service on NS representing DNS server DNS-2	SVC-DNS-2	• IP address = 203.0.113.60 • Port = 53
DNS64 action	DNS64-Action-1	• DNS64 Prefix=2001:DB8:300::
DNS64 policy	DNS64-Policy-1	• DNS64 action = DNS64-Action-1 • Rule= CLIENT.IP.SRC.IN_SUBNET(2001:DB8:5001::/64)
DNS load balancing virtual server	LBVS-DNS64-1	• IP address=2001:DB8:9999::99 • Bound DNS services= SVC-DNS-1, SVC-DNS-2 • DNS64=Enabled • Bound DNS64 policy= DNS64-Policy-1

Following is the traffic flow in this example:

1. IPv6 client CL1 sends a DNS AAAA request for the IPv6 address of the site www.example.com.
2. The request is received by the DNS load balancing virtual server LBVS-DNS64-1 on NetScaler appliance NS1.
3. NS1 checks its DNS cache records for the requested AAAA record and finds that AAAA record for the site www.example.com does not exist in the DNS cache.
4. LBVS-DNS64-1's load balancing algorithm selects DNS server DNS-1 and forwards the AAAA request to it.
5. Because the site www.example.com is hosted on an IPv4 server, the DNS server DNS-1 does not have any AAAA record for the site www.example.com.
6. DNS-1 sends either an empty DNS AAAA response or an error message to LBVS-DNS64-1.
7. Because DNS64 option is enabled on LBVS-DNS64-1 and the AAAA request from CL1 matches the condition specified in DNS64-Policy-1, NS1 sends a DNS A request to DNS-1 for the IPv4 address of www.example.com.
8. DNS-1 responds by sending the DNS A record for www.example.com to LBVS-DNS64-1. The A record includes the IPv4 address for www.example.com.
9. NS1 synthesizes an AAAA record for the site www.example.com with:
   - IPv6 address for site www.example.com = Concatenation of DNS64 Prefix (96 bits) specified in the associated DNS64action, and IPv4 address of DNS A record (32 bits) = 2001:DB8:300::192.0.2.60
10. NS1 sends the synthesized AAAA record to IPv6 client CL1. NS1 also caches the A record into its memory. NS1 uses the cached A record to synthesize AAAA records for subsequent AAAA requests.

This section includes the following details:

- Points to Consider for a DNS64 Configuration
- Configuration Steps

Points to Consider for a DNS64 Configuration

Before configuring DNS64 on a NetScaler appliance, consider the following points:

- The DNS64 feature of the NetScaler appliance is compliant with RFC 6174.
- The DNS64 feature of the NetScaler appliance does not support DNSSEC. The NetScaler appliance does not synthesize an AAAA record from a DNSSEC response received from a DNS server. A response is classified as a DNSSEC response, only if it contains RRSIG records.

- The NetScaler appliance supports DNS64 prefix of length of only 96 bits.
- Though the DNS64 feature is used with the NAT64 feature, the DNS64 and NAT64 configurations are independent on the NetScaler appliance. For a particular flow, you must specify the same IPv6 prefix value for the DNS64 prefix and the NAT64 prefix parameters, so that the synthesized IPv6 addresses received by the client are routed to the particular NAT64 configuration. For more information on configuring NAT64 on a NetScaler appliance, see "Stateful NAT64."
- The following are the different cases of DN64 processing by the NetScaler appliance:
  - If the AAAA response from the DNS server includes AAAA records, then each record in the response is checked for the set of exclusion rule configured on the NetScaler appliance for the particular DNS64 configuration. The NetScaler removes the IPv6 addresses, whose prefix matches the exclusion rule, from the response. If the resulting response includes at least one IPv6 record, the NetScaler appliance forwards this response to the client, else, the appliance synthesizes a AAAA response from the A record of the domain and sends it to the IPv6 client.
  - If the AAAA response from the DNS server is an empty answer response, the appliance requests for A resource records with the same domain name or searches in its own records if the appliance is an authentic domain name server for the domain. If the request results in an empty answer or error, the same is forwarded to the client.
  - If the response from the DNS server includes RCODE=1 (format error), the NetScaler appliance forwards the same to the client. If there is no response before the timeout, the NetScaler appliance sends a response with RCODE=2 (server failure) to the client.
  - If the response from the DNS server includes a CNAME, the chain is followed until the terminating A or AAAA record is reached. If the CNAME does not have any AAAA resource records, the NetScaler appliance fetches the DNS A record to be used for synthesizing AAAA record. The CNAME chain is added to the answer section along with the synthesized AAAA record and then sent to the client.
- The DNS64 feature of the NetScaler appliance also supports responding to PTR request. When a PTR request for a domain of an IPv6 address is received on the appliance and the IPv6 address matches any of the configured DNS64 prefix, the appliance creates a CNAME record mapping the IP6-ARPA domain into the corresponding IN-ADDR.ARPA domain and the newly formed IN-ADDR.ARPA domain is used for resolution. The appliance searches the local PTR records and if the records are not present, the appliance sends a PTR request for IN-ADDR.ARPA domain to the DNS server. The NetScaler appliance uses the response from the DNS server to synthesize response for the initial PTR request.

Configuration Steps

Updated: 2013-09-30

Creating the required entities for stateful NAT64 configuration on the NetScaler appliance involves the following procedures:

- **Add DNS services.** DNS services are logical representation of DNS servers for which the NetScaler appliance acts as a DNS proxy server. For more information on setting optional parameters of a service, see "Load Balancing".
- **Add DNS64 action and DNS64 policy and then bind the DNS64 action to the DNS64 policy.** A DNS64 policy specifies conditions to be matched against traffic for DNS64 processing according to the settings in the associated DNS64 action. The DNS64 action specifies the mandatory DNS64 prefix and the optional exclude rule and mapped rule settings.
- **Create a DNS load balancing virtual server and bind the DNS services and the DNS64 policy to it.** The DNS load balancing virtual server acts as a DNS proxy server for DNS servers represented by the bound DNS services. Traffic arriving at the virtual server is matched against the bound DNS64 policy for DNS64 processing. For more information on setting optional parameters of a load balancing virtual server, see "Load Balancing".
  Note: The command line interface has separate commands for these two tasks, but the configuration utility combines them in a single dialog box.

- **Enable caching of DNS records.** Enable the global parameter for the NetScaler appliance to cache DNS records, which are obtained through DNS proxy operations. For more information on enabling caching of DNS records, see "Enabling Caching of DNS Records".

To create a service of type DNS by using the command line interface

At the command prompt, type:

- add service <name> <IP> <serviceType> <port> ...

To create a DNS64 action by using the command line interface

At the command prompt, type:

- add dns action64 <actionName> -Prefix <ipv6_addr|*> [-mappedRule <expression>] [-excludeRule <expression>]

To create a DNS64 policy by using the command line interface

At the command prompt, type:

- add dns policy64 <name> -rule <expression> -action <string>

To create a DNS load balancing virtual server by using the command line interface

At the command prompt, type:

- add lb vserver <name> DNS <IPAddress> <port> -dns64 (ENABLED | DISABLED) [-bypassAAAA (YES | NO)] ...

To bind the DNS services and the DNS64 policy to the DNS load balancing virtual server by using the command line interface

At the command prompt, type:

- bind lb vserver <name> <serviceName> ...
- bind lb vserver <name> -policyName <string> -priority <positive_integer> ...

Example

```
 > add service SVC-DNS-1 203.0.113.50 DNS 53
 Done

> add service SVC-DNS-2 203.0.113.60 DNS 53
 Done

> add dns Action64 DNS64-Action-1 -Prefix 2001:DB8:300::/96
 Done

> add dns Policy64 DNS64-Policy-1 -rule "CLIENT.IPv6.SRC.IN_SUBNET(2001:DB8:5001::/64)"
-action DNS64-Action-1
 Done

> add lb vserver LBVS-DNS64-1 DNS 2001:DB8:9999::99 53 -dns64 ENABLED
 Done

> bind lb vserver LBVS-DNS64-1 SVC-DNS-1
 Done

> bind lb vserver LBVS-DNS64-1 SVC-DNS-2
 Done
```

```
> bind lb vserver LBVS-DNS64-1 -policyname DNS64-Policy-1 -priority 2
 Done
```

To create a service of type DNS by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services, and add a new service.
2. Set the following parameters:
   - Service Name*
   - Server*
   - Protocol* (Select DNS from the drop down list.)
   - Port*

To create a DNS64 action by using the configuration utility

Navigate to Traffic Management > DNS > Actions, on the DNS Actions64 tab, add a new DNS64 action.

To create a DNS64 policy by using the configuration utility

Navigate to Traffic Management > DNS > Policies, on the DNS Policies64 tab, add a new DNS64 policy.

To create a DNS load balancing virtual server and bind the DNS services and the DNS64 policy to it by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers, and add a new virtual server.
2. Set the following parameters:
   - Name*
   - IP Address*
   - Protocol* (Select DNS from the drop down list.)
   - Port*
3. Select the Enable DNS64 option.
4. In the Services pane, bind the service to the virtual server.
5. In the Policies pane, bind the policy to the virtual server.

Stateful NAT64 Translation

May 11, 2012

The stateful NAT64 feature enables communication between IPv6 clients and IPv4 servers through IPv6 to IPv4 packet translation, and vice versa, while maintaining session information on the NetScaler appliance.

A stateful NAT64 configuration on the NetScaler appliance has the following components:

- **NAT64 rule**— An entry consisting of an ACL6 rule and a netprofile, which consists of a pool of NetScaler owned SNIP Addresses.
- **NAT64 IPv6 Prefix**— A global IPv6 prefix of length 96 bits (128-32=96) configured on the appliance.
  Note: Currently the NetScaler appliance supports only one prefix to be used commonly with all NAT 64 rules.

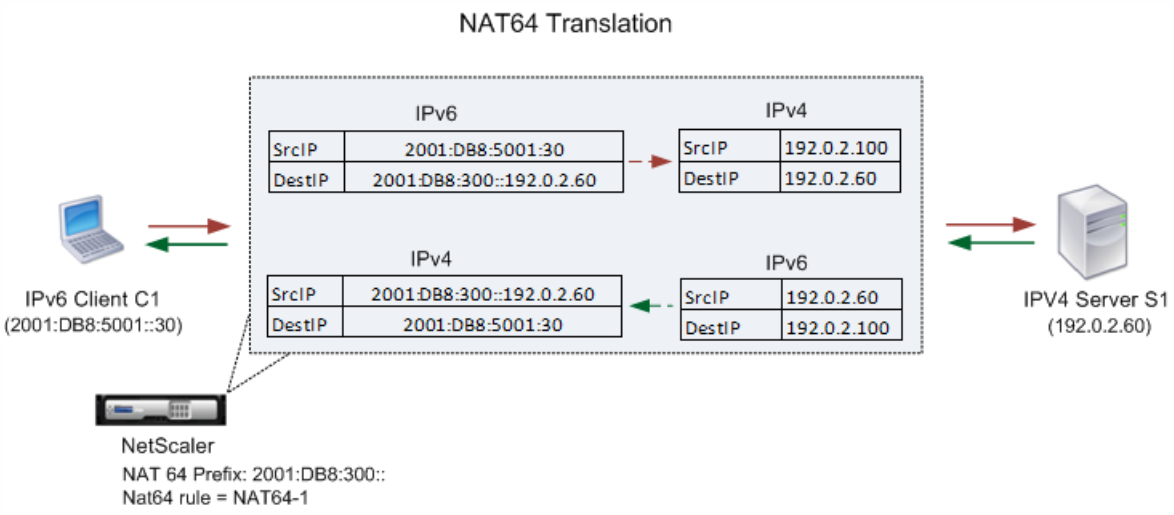
The NetScaler appliance considers an incoming IPv6 packet for NAT64 translation when all of the following conditions are met:

- The incoming IPv6 packet matches the ACL6 rule bound to a NAT64 rule.
- The destination IP address of the IPv6 packet matches the NAT64 IPv6 prefix.

When an IPv6 request packet received by the NetScaler appliance matches an ACL6 defined in a NAT64 rule and the destination IP of the packet matches the NAT64 IPv6 prefix, the NetScaler appliance considers the IPv6 packet for translation.

The appliance translates this IPv6 packet to an IPv4 packet with a source IP address matching one of the IP address bound to the netprofile defined in the NAT64 rule, and a destination IP address consisting of the last 32 bits of the destination IPv6 address of the IPv6 request packet. The NetScaler appliance creates a NAT64 session for this particular flow and forwards the packet to the IPv4 server. Subsequent responses from the IPv4 server and requests from the IPv6 client are translated accordingly by the appliance, on the basis of information in the particular NAT64 session.

Consider an example in which an enterprise hosts site www.example.com on server S1, which has an IPv4 address. To enable communication between IPv6 clients and IPv4 server S1, NetScaler appliance NS1 is deployed with a stateful NAT64 configuration that includes a NAT64 rule and a NAT64 prefix. A mapped IPv6 address of server S1 is formed by concatenating the NAT64 IPv6 prefix [96 bits] and the IPv4 source address [32 bits]. This mapped IPv6 address is then manually configured in the DNS servers. The IPv6 clients get the mapped IPv6 address from the DNS servers to communicate withIPv4 server S1.



The following table lists the settings used in this example:

Entities	Name	Value
IPv6 address of client CL1	Client_IPv6 (for reference purposes only)	2001:DB8:5001::30
IPv4 address of server S1	Sevr_IPv4 (for reference purposes only)	192.0.2.60
IPv6 prefix for NAT64 translation	NAT64_Prefix (for reference purposes only)	2001:DB8:300::
Mapped IPv6 address (NAT64_Prefix + Sevr_IPv4) of server S1 for IPv6 clients to reach server S1	Map-Sevr-IPv6 (for reference purposes only)	2001:DB8:300::192.0.2.60
ACL6 rule	ACL6-1	• Action = ALLOW • Source IP address =2001:DB8:5001::30
IPset	IPset-1	IP addresses bound (of type SNIPs) = 192.0.2.100 and 192.0.2.102
Netprofile	Netprofile-1	Source IP address = IPset-1
NAT64 rule	NAT64-1	ACL6 rule = ACL6-1 Netprofile = Netprofile-1

Following is the traffic flow in this example:

1. IPv6 client CL1 sends a request packet to Map-Sevr-IPv6 (2001:DB8:300::192.0.2.60) address.
2. The NetScaler appliance receives the request packet. If the request packet matches the ACL6 defined in the NAT64 rule, and the destination IP address of the packet matches the NAT64 IPv6 prefix, the NetScaler considers the IPv6 packet for translation.
3. The appliance creates a translated IPv4 request packet with:
   - Destination IP address field containing the NAT64 prefix stripped from the destination address of the IPv6 request (Sevr_IPv4 = 192.0.2.60)
   - Source IP address field containing one of the IPv4 address bound to Netprofile-1(in this case, 192.0.2.100)
4. The NetScaler appliance creates a NAT64 session for this flow and sends the translated IPv4 request to server S1.
5. IPv64 server S1 responds by sending an IPv4 packet to the NetScaler appliance with:
   - Destination IP address field containing 192.0.2.100
   - Source IP address field containing the address ofSevr_IPv4(192.0.2.60)
6. The appliance receives the IPv4 response packet, searches all the session entries, and finds that the IPv6 response packet matches the NAT64 session entry created in step 4. The appliance considers the IPv4 packet for translation.

7. The appliance creates a translated IPv6 response packet with:
   - Destination IP address field=Client_IPv6=2001:DB8:5001::30
   - Source IP address field = Concatenation of NAT64 Prefix (First 96 bits) and Sevr_IPv4 (last 32 bits) =2001:DB8:300::192.0.2.60
8. The appliance sends the translated IPv6 response to client CL1.

This section includes the following details:

- Limitations of Statelful NAT64
- Configuring Stateful NAT64

Limitations of Statelful NAT64

The following limitations apply to stateful NAT64 translation:

- Translation of IPv4 options is not supported.
- Translation of IPv6 routing headers is not supported.
- Translation of hop-by-hop extension headers of IPv6 packets is not supported.
- Translation of ESP and EH headers of IPv6 packets is not supported.
- Translation of multicast packets is not supported.
- Packets of Stream Control Transmission Protocol (SCTP), Datagram Congestion Control Protocol (DCCP), and IPSec, are not translated.

Configuring Stateful NAT64

Updated: 2013-10-31

Creating the required entities for stateful NAT64 configuration on the NetScaler appliance involves the following procedures:

1. Add an ACL6 rule with action ALLOW.
2. Add an ipset, which binds multiple IP addresses.
3. Add a netprofile and bind the ipset to it. If you want to bind only one IP address, you need not create an ipset entity. In that case, bind the IP address directly to the netprofile.
4. Add a NAT64 rule, which includes binding the ACl6 rule and the netprofile to the NAT 64 rule.
5. Add a NAT64 IPv6 prefix.

To add an ACL6 rule by using the command line interface

At the command prompt, type:

- add ns acl6 <acl6name> <acl6action> ...

To add an IPset and bind multiple IPs to it by using the command line interface

At the command prompt, type:

- add ipset <name>
- bind ipset <name> <IPaddress ...>

To add a netprofile by using the command line interface

At the command prompt, type:

- add netprofile <name> -srcIP <IPaddress or IPset>

To add a NAT64 rule by using the command line interface

At the command prompt, type:

- add nat64 <name> <acl6name> -netProfile <string>

To add a NAT64 prefix by using the command line interface

At the command prompt, type:

- set ipv6 -natprefix <ipv6_addr|*>

Example

```
 > add acl6 ACL6-1 ALLOW -srcIPv6 2001:DB8:5001::30
 Done

> apply acls6
 Done

> add ip 192.0.2.100 255.255.255.0 –type SNIP
 Done

> add ip 192.0.2.102 255.255.255.0  –type SNIP
 Done

> add ipset IPset-1
 Done

> bind ipset IPset-1 192.0.2.100 192.0.2.102
IPAddress "192.0.2.100" bound
IPAddress "192.0.2.102" bound
 Done

> add netprofile Netprofile-1 -srcIP IPset-1
 Done

> add nat64 NAT64-1 ACL6-1 -netprofile Netprofile-1
 Done

> set ipv6 -natprefix 2001:DB8:300::/96
 Done
```

To add a NAT64 rule by using the configuration utility

Navigate to System > Network > Routes > NAT64, and a new NAT64 rule, or edit an existing rule.

To add a NAT64 prefix by using the configuration utility

Navigate to System > Network, in the Settings group, click Configure INAT Parameters, and set the Prefix parameter.

Configuring RNAT

May 11, 2012

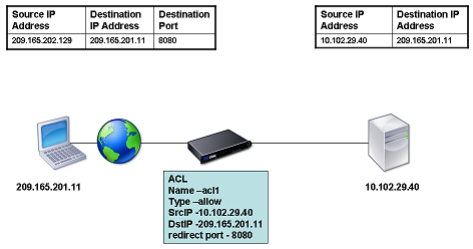
In Reverse Network Address Translation (RNAT), the NetScaler appliance replaces the source IP addresses in the packets generated by the servers with public NAT IP addresses. By default, the appliance uses a Mapped IP address (MIP) as the NAT IP address. You can also configure the appliance to use a unique NAT IP address for each subnet. You can also configure RNAT by using Access Control Lists (ACLs). Use Source IP (USIP), Use Subnet IP (USNIP), and Link Load Balancing (LLB) modes affect the operation of RNAT. You can display statistics to monitor RNAT.

Note: The ephemeral port range for RNAT on the NetScaler appliance is 1024-65535.
You can use either a network address or an extended ACL as the condition for an RNAT entry:

- **Using a Network address.** When you use a network address, RNAT processing is performed on all of the packets coming from the specified network.
- **Using Extended ACLs.** When you use ACLs, RNAT processing is performed on all packets that match the ACLs. To configure the NetScaler appliance to use a unique IP address for traffic that matches an ACL, you must perform the following three tasks:
  1. Configure the ACL.
  2. Configure RNAT to change the source IP address and Destination Port.
  3. Apply the ACL.

  The following diagram illustrates RNAT configured with an ACL.

  Figure 1. RNAT with an ACL

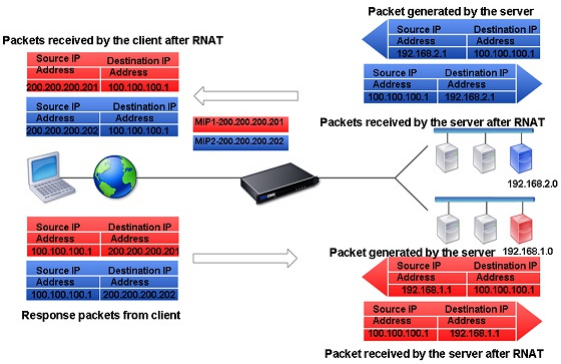


You have the following basic choices for the type of NAT IP address:

- **Using a MIP or SNIP as the NAT IP Address.** When using a MIP as the NAT IP address, the NetScaler appliance replaces the source IP addresses of server-generated packets with the a MIP. Therefore, the MIP address must be a public IP address. If Use Subnet IP (USNIP) mode is enabled, the NetScaler can use a subnet IP address (SNIP) as the NAT IP address.
- **Using a Unique IP Address as the NAT IP Address.** When using a unique IP address as the NAT IP address, the NetScaler appliance replaces the source IP addresses of server-generated packets with the unique IP address specified. The unique IP address must be a public NetScaler-owned IP address. If multiple NAT IP addresses are configured for a subnet, NAT IP selection uses the round robin algorithm.

  This configuration is illustrated in the following diagram.

  Figure 2. Using a Unique IP Address as the NAT IP Address



This section includes the following details:

- Creating an RNAT Entry
- Monitoring RNAT
- RNAT in USIP, USNIP, and LLB Modes
- Configuring RNAT for IPv6 Traffic

Creating an RNAT Entry

Updated: 2013-08-28

The following instructions provide separate command-line procedures for creating RNAT entries that use different conditions and different types of NAT IP addresses. In the configuration utility, all of the variations can be configured in the same dialog box, so there is only one procedure for configuration utility users.

To create an RNAT entry by using the command line interface

At the command prompt, type one the following commands to create, respectively, an RNAT entry that uses a network address as the condition and a MIP or SNIP as the NAT IP address, an RNAT entry that uses a network address as the condition and a unique IP address as the NAT IP address, an RNAT entry that uses an ACL as the condition and a MIP or SNIP as the NAT IP address, or an RNAT entry that uses an ACL as a condition and a unique IP address as the NAT IP address:

- set rnat <IPAddress> <netmask>
- set rnat IPAddress <netMask> -natip <NATIPAddress>
- set rnat <aclname> [-redirectPort <port>]
- set rnat <aclname> [-redirectPort <port>] -natIP <NATIPAddress>

Use the following command to verify the configuration:

- show rnat

Examples

A network address as the condition and a MIP or SNIP as the NAT IP address:

```
> set rnat 192.168.1.0 255.255.255.0
 Done
```

A network address as the condition and a unique IP address as the NAT IP address:

```
> set rnat 192.168.1.0 255.255.255.0 -natip 10.102.29.50
 Done
```

If instead of a single NAT IP address you specify a range, RNAT entries are created with all the NetScaler-owned IP addresses, except the NSIP, that fall within the range sp

```
> set rnat 192.168.1.0 255.255.255.0  -natIP 10.102.29.[50-110]
 Done
```

An ACL as the condition and a MIP or SNIP as the NAT IP address:

```
> set rnat acl1
 Done
```

An ACL as a condition and a unique IP address as the NAT IP address:

```
> set rnat acl1  -natIP 209.165.202.129
Done
```

If instead of a single NAT IP address you specify a range, RNAT entries are created with all the NetScaler-owned IP addresses, except the NSIP, that fall within the range sp

```
> set rnat acl1  -natIP 10.102.29.[50-70]
 Done
```

To create an RNAT entry by using the configuration utility

1. Navigate to System > Network > Routes > RNAT.
2. In the Action list, select Configure RNAT.

Monitoring RNAT

Updated: 2013-09-27

You can display RNAT statistics to troubleshoot issues related to IP address translation.

To view RNAT statistics by using the command line interface

At the command prompt, type:

stat rnat

Example

```
> stat rnat

RNAT summary
                        Rate (/s)          Total
Bytes Received                0                0
Bytes Sent                    0                0
Packets Received              0                0
Packets Sent                  0                0
Syn Sent                      0                0
Current RNAT sessions         --               0
 Done
>
```

The following tables describes the statistics associated with RNAT and RNAT IP.

Table 1. RNAT Statistics

Statistic	Description
Bytes received	Bytes received during RNAT sessions

Statistic	Description
Bytes sent	Bytes sent during RNAT sessions
Packets received	Packets received during RNAT sessions
Packets sent	Packets sent during RNAT sessions
Syn sent	Requests for connections sent during RNAT sessions
Current sessions	Currently active RNAT sessions

To monitor RNAT by using the configuration utility

Navigate to System > Network > Routes > RNAT, and click Statistics.

RNAT in USIP, USNIP, and LLB Modes

Updated: 2013-12-18

Before configuring a RNAT rule, consider the following points:

- When RNAT and Use Source IP (USIP) are both configured on the NetScaler appliance, RNAT takes precedence. In other words, the source IP address of the packets, which matches a RNAT rule, is replaced according to the setting in the RNAT rule.
- When RNAT and Use SNIP (USNIP) are configured on the NetScaler appliance, selection of the source IP address is based on the state of USNIP, as follows:
  - If USNIP is off, the NetScaler appliance uses the mapped IP addresses.
  - If USNIP is on, the NetScaler uses a SNIP address as the NAT IP address.

This behavior does not apply when a unique NAT IP address is used.

In a topology where the NetScaler appliance performs both Link Load Balancing (LLB) and RNAT for traffic originating from the server, the appliance selects the source IP address based on the router. The LLB configuration determines selection of the router. For more information about LLB, see "Link Load Balancing."

Configuring RNAT for IPv6 Traffic

Updated: 2013-10-31

Reverse Network Address Translation (RNAT) rules for IPv6 packets are called RNAT6s. When an IPv6 packet generated by a server matches the conditions specified in the RNAT6 rule, the appliance replaces the source IPv6 address of the IPv6 packet with a configured NAT IPv6 address before forwarding it to the destination. The NAT IPv6 address is one of the NetScaler owned SNIP6 or VIP6 addresses.

When configuring an RNAT6 rule, you can specify either an IPv6 prefix or an ACL6 as the condition:

- **Using a IPv6 network address**. When you use an IPv6 prefix, the appliance performs RNAT processing on those IPv6 packets whose IPv6 address matches the prefix.
- **Using ACL6s**. When you use an ACL6, the appliance performs RNAT processing on those IPv6 packets that match the conditions specified in the ACL6.

You have one of the following options to set the NAT IP address:

- Specify a set of NetScaler owned SNIP6 and VIP6 addresses for an RNAT6 rule. The NetScaler appliance uses any one of the IPv6 addresses from this set as a NAT IP address for each session. The selection is based on the round robin algorithm and is done for each session.
- Do not specify any NetScaler owned SNIP6 or VIP6 address for an RNAT6 rule. The NetScaler appliance uses any one of the NetScaler owned SNIP6 or VIP6 addresses as a NAT IP address. The selection is based on the next hop network to which an IPv6 packet that matches the RNAT rule is destined.

To create an RNAT6 rule by using the command line interface

At the command prompt, to create the rule and verify the configuration, type:

- add rnat6 <name> (<network> | (<acl6name> [-redirectPort <port>]))
- bind rnat6 <name> <natIP6>@ ...
- show rnat6

To modify or remove an RNAT6 rule by using the command line interface

- To modify an RNAT6 rule whose condition is an ACL6, type the set rnat6 <name> command, followed by a new value for the redirectPort parameter.
- To remove an RNAT6 rule, type the clear rnat6 <name> command.
- show rnat6

To configure an RNAT6 rule by using the configuration utility

Navigate to System > Network > Routes > RNAT6, and add a new RNAT6 rule, or edit an existing rule.

Configuring Prefix-Based IPv6-IPv4 Translation

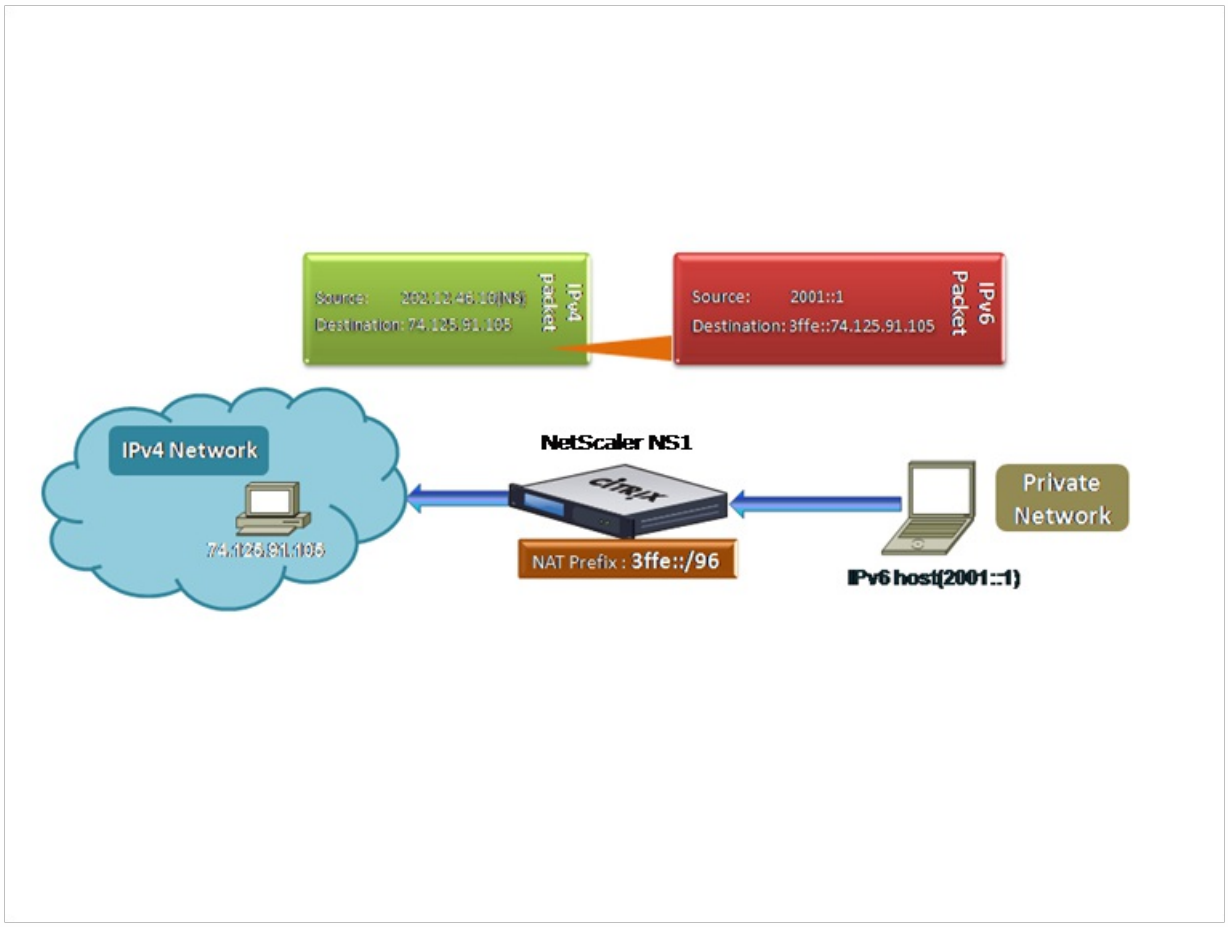
Aug 28, 2013

Prefix-based translation is a process of translating packets sent from private IPv6 servers into IPv4 packets, using an IPv6 prefix configured in the NetScaler appliance. This prefix has a length of 96 bits (128-32=96). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix.

The NetScaler appliance compares the first 96 bits of the destination IP address of all the incoming IPv6 packets to the configured prefix. If there is a match, the NetScaler appliance generates an IPv4 packet and sets the destination IP address as the last 32 bits of the destination IP address of the matched IPv6 packet. IPv6 packets addressed to this prefix have to be routed to the NetScaler so that the IPv6-IPv4 translation is done by the NetScaler.

In the following diagram, 3ffe::/96 is configured as the IPv6 NAT prefix on NetScaler NS1. The IPv6 host sends an IPv6 packet with destination IP address 3ffe::74.125.91.105. NS1 compares the first 96 bits of the destination IP address of all the incoming IPv6 packets to the configured prefix, and they match. NS1 then generates an IPv4 packet and sets the destination IP address as 74.125.91.105.

Figure 1. IPv6-IPv4 Prefix-Based Translation



To configure prefix-based IPv6-IPv4 translation by using the command line interface

At the command prompt, type the following commands to set a NAT prefix and verify its configuration:

- set ipv6 [-natprefix <ipv6_addr|*>]
- show ipv6

Example

```
> set ipv6 -natprefix 3ffe::/96
 Done
```

To configure prefix-based IPv6-IPv4 translation by using the configuration utility

Navigate to System > Network, in the Settings group, click Configure INAT Parameters, and set the Prefix parameter.

Configuring Static ARP

Aug 28, 2013

You can add static ARP entries to and remove static ARP entries from the ARP table. After adding an entry, you should verify the configuration. If the IP address, port, or MAC address changes after you create a static ARP entry, you must remove or manually adjust the static entry. Therefore, creating static ARP entries is not recommended unless necessary.

To add a static ARP entry by using the command line interface

At the command prompt, type:

- add arp -IPAddress <ip_addr> -mac<mac_addr> -ifnum <interface_name>
- show arp <IPAddress>

Example

```
> add arp -ip 10.102.29.6 -mac 00:24:e8:73:ca:ec -ifnum 1/1
 Done
```

To remove a static ARP entry by using the command line interface

At the command prompt, type the rm arp command and the IP address.

To add a static ARP entry by using the configuration utility

Navigate to System > Network > ARP Table, and a new ARP entry.

Setting the Timeout for Dynamic ARP Entries

Aug 28, 2013

You can globally set an aging time (time-out value) for dynamically learned ARP entries. The new value applies only to ARP entries that are dynamically learned after the new value is set. Previously existing ARP entries expire after the previously configured aging time.

You can specify an ARP time-out value of from 1 through 1200 seconds.

To set the time-out for dynamic ARP entries by using the command line interface

At the command prompt, type the following commands to set the time-out for dynamic ARP entries and verify its configuration:

- set arpparam -timeout <positive_integer>]
- show arpparam

Example

```
> set arpparam -timeout 500
 Done
```

To set the time-out for dynamic ARP entries to its default value by using the command line interface

At the command prompt, type the following commands to set the time-out for dynamic ARP entries to its default value and verify its configuration:

- unset arpparam
- show arpparam

Example

```
> unset arpparam
 Done
```

To set the time-out for dynamic ARP entries by using the configuration utility

Navigate to System > Network, in the Settings group, click Configure ARP Global Parameters, and set the ARP Table Entry Timeout parameter.

Configuring Neighbor Discovery

May 10, 2012

Neighbor discovery (ND) is one of the most important protocols of IPv6. It is a message-based protocol that combines the functionality of the Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), and Router Discovery. ND allows nodes to advertise their link layer addresses and obtain the MAC addresses or link layer addresses of the neighboring nodes. This process is performed by the Neighbor Discovery protocol (ND6).

Neighbor discovery can perform the following functions:

Router Discovery

Enables a host to discover the local routers on an attached link and automatically configure a default router.

Prefix Discovery

Enables the host to discover the network prefixes for local destinations.
Note: Currently, the NetScaler does not support Prefix Discovery.

Parameter Discovery

Enables a host to discover additional operating parameters, such as MTU and the default hop limit for outbound traffic.

Address Autoconfiguration

Enables hosts to automatically configure IP addresses for interfaces both with and without stateful address configuration services such as DHCPv6. The NetScaler does not support Address Autoconfiguration for Global IPv6 addresses.

Address Resolution

Equivalent to ARP in IPv4, enables a node to resolve a neighboring node's IPv6 address to its link-layer address.

Neighbor Unreachability Detection

Enables a node to determine the reachability state of a neighbor.

Duplicate Address Detection

Enables a node to determine whether an NSIP address is already in use by a neighboring node.

Redirect

Equivalent to the IPv4 ICMP Redirect message, enables a router to redirect the host to a better first-hop IPv6 address to reach a destination.

Note: The NetScaler does not support IPv6 Redirect.
To enable neighbor discovery, you create entries for the neighbors.

This section includes the following details:

- Adding IPv6 Neighbors
- Removing IPv6 Neighbors

Adding IPv6 Neighbors

Updated: 2013-08-28

Adding IPv6 neighbors enables neighbor discovery.

To add an IPv6 neighbor by using the command line interface

At the command prompt, type:

- add nd6 <neighbor> <mac> <ifnum> [-vlan <integer>]
- show nd6

Example

```
> add nd6 2001::1 00:04:23:be:3c:06 1/1 –vlan 1
 Done
> show nd6
   Neighbor             MAC-Address(Vlan, Interface)      State       TIME
   --------             ----------------------------      -----     --------
1) ::1                   00:d0:68:0b:58:da(   1,  LO/1)   REACHABLE    PERMANENT
2) fe80::2d0:68ff:fe0b:58da 00:d0:68:0b:58:da(   1,  LO/1)   REACHABLE    PERMANENT
3) 2001::1               00:04:23:be:3c:06(   1,   1/1)   REACHABLE      STATIC
 Done
```

To add an IPv6 neighbor by using the configuration utility

Navigate to System > Network > IPv6 Neighbors, and add a new IPv6 neighbor.

Removing IPv6 Neighbors

Updated: 2013-08-28

To remove a neighbor discovery entry by using the command line interface

At the command prompt, type:

rm nd6 <Neighbor> -vlan <VLANID>

Example

```
 rm nd6 3ffe:100:100::1 -vlan 1
```

To remove all neighbor discovery entries by using the command line interface

At the command prompt, type:

clear nd6

To remove a neighbor discovery entry by using the configuration utility

Navigate to System > Network > IPv6 Neighbors, delete the IPv6 neighbor.

To remove all neighbor discovery entries by using the configuration utility

Navigate to System > Network > IPv6 Neighbors, and click Clear.

Configuring IP Tunnels

Mar 20, 2012

An IP Tunnel is a communication channel, that can be created by using encapsulation technologies, between two networks that do not have a routing path. Every IP packet that is shared between the two networks is encapsulated within another packet and then sent via the tunnel.

The NetScaler appliance implements IP Tunneling in the following ways:

- NetScaler as an Encapsulator (Load Balancing with DSR mode)
- NetScaler as a Decapsulator

NetScaler as an Encapsulator (Load Balancing with DSR Mode)

Consider an organization that has multiple data centers across different countries, where the NetScaler maybe at one location and the back-end servers are located in a different country. In essence, the NetScaler and the back-end servers are on different networks and are connected via a router.

When you configure Direct Server Return (DSR) on this NetScaler, the packet sent from the source subnet is encapsulated by the NetScaler and sent via a router and tunnel to the appropriate back-end server. The back-end server decapsulates the packet and responds directly to the client, without allowing the packet to pass via the NetScaler.

NetScaler as a Decapsulator

Consider an organization having multiple data centers each having NetScalers and back-end servers. When a packet is sent from data center A to data center B it is usually sent via an intermediary, say a router or another NetScaler. The NetScaler processes the packet and then forwards the packet to the back-end server. However, if an encapsulated packet is sent, the NetScaler must be able to decapsulate the packet before sending it to the back-end servers. To enable the NetScaler to function as a decapsulator, a tunnel is added between the router and the NetScaler. When the encapsulated packet, with additional header information, reaches the NetScaler, the data packet is decapsulated i.e. the additional header information is removed, and the packet is then forwarded to the appropriate back-end servers.

The NetScaler can also be used as a decapsulator for the Load Balancing feature, specifically in scenarios when the number of connections on a vserver exceeds a threshold value and all the new connections are then diverted to a back-up vserver.

This section includes the following details:

- Creating IP Tunnels
- Customizing IP Tunnels Globally

Creating IP Tunnels

Updated: 2013-10-31

To create an IP tunnel by using the command line interface

At the command prompt type:

- add iptunnel <name> <remote> <remoteSubnetMask> <local> -type -protocol (ipoverip | GRE) -ipsecprofile <name>
- show iptunnel

Note: While configuring an IP tunnel in a cluster setup, the local IP address must be a striped SNIP or MIP address.

To remove an IP tunnel by using the command line interface

To remove an IP tunnel, type the rm iptunnel command and the name of the tunnel.

To create an IP Tunnel by using the configuration utility

Navigate to System > Network > IP Tunnels, add a new IP tunnel.

To create an IPv6 tunnel by using the command line interface

At the command prompt type:

- add ip6tunnel <name> <remoteIp> <local>
- show ip6tunnel

To remove an IPv6 tunnel by using the command line interface

To remove an IPv6 tunnel, type the rm ip6tunnel command and the name of the tunnel.

To create an IPv6 Tunnel by using the configuration utility

Navigate to System > Network > IP Tunnels > IPv6 Tunnels, and add a new IPv6 tunnel.

Customizing IP Tunnels Globally

Updated: 2013-10-31

By globally specifying the source IP address, you can assign a common source IP address across all tunnels. Also, because fragmentation is CPU-intensive, you can globally specify that the NetScaler appliance drop any packet that requires fragmentation. Alternatively, if you would like to fragment all packets as long as a CPU threshold value is not reached, you can globally specify the CPU threshold value.

To globally customize IP tunnels by using the command line interface

At the command prompt, type the following commands to globally customize IP tunnels and verify the configuration:

- set ipTunnelParam -srcIP <sourceIPAddress> -srcIPRoundRobin (YES | NO)-dropFrag [YES | NO] -dropFragCpuThreshold <Positive integer>
- show ipTunnelParam

Example

```
> set iptunnelparam –srcIP 12.12.12.22 -dropFrag Yes –dropFragCpuThreshold 50
 Done
> set iptunnelparam -srcIPRoundRobin YES -dropFrag Yes –dropFragCpuThreshold 50
 Done
```

Note: To create a new MIP or SNIP address to use as the global source IP address, use the add ns ip command before you type the set iptunnelparam command.

To globally customize IP tunnels by using the configuration utility

Navigate to System > Network, in the Settings group, click IPv4 Tunnel Global Settings.

1. Navigate to System > Network.
2. In the details pane, in the Settings group, click IPv4 Tunnel Global Settings.
3. In the Configure IP Tunnel Global Parameters dialog box, set the parameters. For a description of a parameter, hover the mouse cursor over the corresponding field.
4. Click OK and then click Close.

To globally customize IPv6 tunnels by using the command line interface

At the command prompt, type the following commands to globally customize IPv6 tunnels and verify the configuration:

- set ip6tunnelparam -srcIP <IPv6Address> -srcIPRoundRobin (YES | NO)-dropFrag [YES | NO] -dropFragCpuThreshold <Positive integer>
- show ip6tunnelparam

Note: To create a new VIP6 or SNIP6 address to use as the global source IP address, use the add ns ip6 command before you type the set ip6tunnelparam command.

To globally customize IPv6 tunnels by using the configuration utility

Navigate to System > Network, in the Settings group, click IPv6 Tunnel Global Settings.

Interfaces

Nov 13, 2015

Before you begin configuring interfaces, decide whether your configuration can use MAC-based forwarding mode, and either enable or disable this system setting accordingly. The number of interfaces in your configuration is different for the different models of the Citrix NetScaler appliance. In addition to configuring individual interfaces, you can logically group interfaces, using VLANs to restrict data flow within a set of interfaces, and you can aggregate links into channels. In a high availability setup, you can configure a virtual MAC (VMAC) address if necessary. If you use L2 mode, you might want to modify the aging of the bridge table.

When your configuration is complete, decide whether you should enable the system setting for path MTU discovery. NetScaler appliances can be deployed in active-active mode using VRRP. An active-active deployment, in addition to preventing downtime, makes efficient use of all the NetScaler appliances in the deployment. You can use the Network Visualizer tool to view the network configuration of a NetScaler deployment and configure interfaces, channels, VLANs, and bridge groups.

This document includes the following information:

- Configuring MAC-Based Forwarding
- Configuring Network Interfaces
- Configuring Forwarding Session Rules
- Understanding VLANs
- Configuring a VLAN
- Configuring NSVLAN
- Configuring Bridge Groups
- Configuring VMACs
- Configuring Link Aggregation
- Binding an SNIP address to an Interface
- Monitoring the Bridge Table and Changing the Aging time
- Understanding NetScaler Appliances in Active-Active Mode Using VRRP
- Configuring Active-Active Mode
- Using the Network Visualizer

Configuring MAC-Based Forwarding

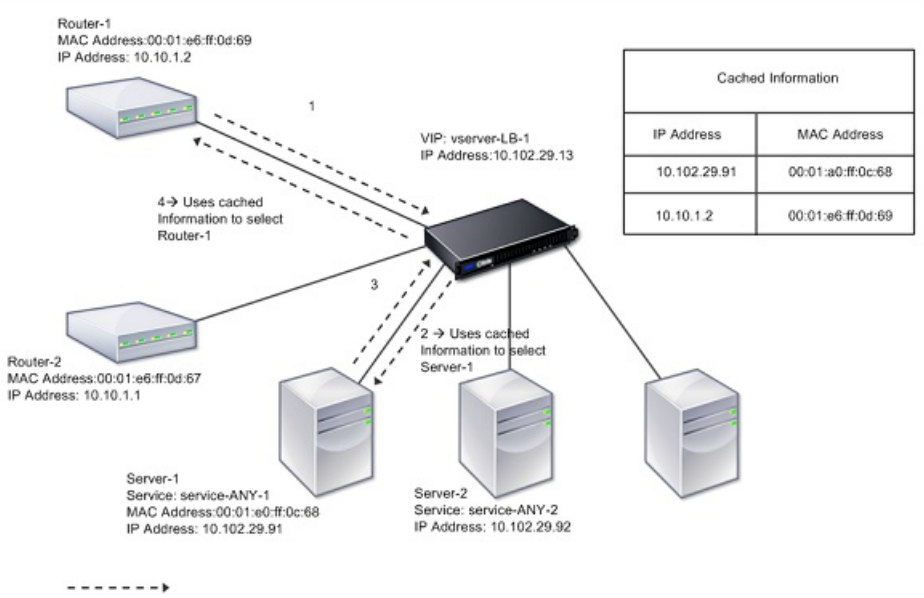
Oct 31, 2013

With MAC-based forwarding (MBF) enabled, when a request reaches the NetScaler appliance, the appliance remembers the source MAC address of the frame and uses it as the destination MAC address for the resulting replies. MAC-based forwarding can be used to avoid multiple-route/ARP lookups and to avoid asymmetrical packet flows. MAC-based forwarding may be required when the NetScaler is connected to multiple stateful devices, such as VPNs or firewalls, because it ensures that the return traffic is sent to the same device that the initial traffic came from.

MAC-based forwarding is useful when you use VPN devices, because it guarantees that all traffic flowing through a VPN passes back through the same VPN device.

The following topology diagram illustrates the process of MAC-based forwarding.

Figure 1. MAC-Based Forwarding Mode



When MAC-based forwarding (MBF) is enabled, the NetScaler caches the MAC address of:

- The source (a transmitting device such as router, firewall, or VPN device) of the inbound connection.
- The server that responds to the requests.

When a server replies through the NetScaler appliance, the appliance sets the destination MAC address of the response packet to the cached address, ensuring that the traffic flows in a symmetric manner, and then forwards the response to the client. The process bypasses the route table lookup and ARP lookup functions. However, when the NetScaler initiates a connection, it uses the route and ARP tables for the lookup function. In a direct server return configuration, you must enable MAC-based forwarding.

For more information about direct server return configurations, see "Load Balancing."

Some deployment topologies may require the incoming and outgoing paths to flow through different routers. MAC-based forwarding would break this topology design.

MBF should be disabled in the following situations:

- **When you configure link load balancing.** In this case, asymmetric traffic flows are desirable because of link costs.
- **When a server uses network interface card (NIC) teaming without using LACP (802.1ad Link Aggregation).** To enable MAC-based forwarding in this situation, you must use a layer 3 device between the NetScaler and server.
  Note: MBF can be enabled when the server uses NIC teaming with LACP, because the virtual interface uses one MAC address.
- When firewall clustering is used. Firewall clustering assumes that ARP is used to resolve the MAC address for inbound traffic. Sometimes the inbound MAC address can be a non-clustered MAC address and should not be used for inbound packet processing.

When MBF is disabled, the NetScaler uses L2 or L3 connectivity to forward the responses from servers to the clients. Depending on the route table, the routers used for outgoing connection and incoming connection can be different. In the case of reverse traffic (response from the server):

- If the source and destination are on different IP subnets, the NetScaler uses the route lookup to locate the destination.
- If the source is on the same subnet as the destination, the NetScaler looks up the ARP table to locate the network interface and forwards the traffic to it. If the ARP table does not exist, the NetScaler requests the ARP entries.

To enable or disable MAC-based forwarding by using the command line interface

At the command prompt, type:

- enable ns mode MBF
- disable ns mode MBF

To enable or disable MAC-based forwarding by using the configuration utility

1. Navigate to System > Settings, in the Modes and Features group, click Configure modes.
2. Select or clear the MAC-based forwarding option.

Configuring Network Interfaces

Nov 13, 2015

Network interfaces in the NetScaler appliance are numbered in <slot>/<port> notation. After configuring your interfaces, you should display the interfaces and their settings to verify the configuration. You can also display this information to troubleshoot a problem in the configuration.

To manage the network interfaces, you might have to enable some interfaces and disable others. You can reset an interface to renegotiate its settings. You can clear the accumulated statistics for an interface. To verify the configuration, you can display the interface settings. You can display the statistics for an interface to evaluate its health.

Setting the Network Interface Parameters

Updated: 2013-09-06

The network interface configuration is neither synchronized nor propagated. For an HA pair, you must perform the configuration on each unit independently.

Network interface parameters include Link Aggregate Control Protocol (LACP) settings. For more information about Link Aggregate Control Protocol (LACP), see "Configuring Link Aggregation Using the Link Aggregate Channel Protocol."

To set the network interface parameters by using the command line interface

At the command prompt, type:

- set interface <id> [-speed <speed>] [-duplex <duplex>] [-flowControl <flowControl>] [-autoneg (DISABLED | ENABLED)] [-haMonitor (ON | OFF)] [(ON | OFF)] [-tagall (ON | OFF)] [-lacpMode <lacpMode>] [-lacpKey<positive_integer>] [-lacpPriority <positive_integer>] [-lacpTimeout (LONG | SHORT)] [-ifAlias <string>] [-throughput <positive_integer>][-bandwidthHigh <positive_integer> [-bandwidthNormal <positive_integer>]]
- show interface [<id>]

Example

```
> set interface 1/8 -duplex full
 Done
```

To set the network interface parameters by using the configuration utility

Navigate to System > Network > Interfaces, select the network interface that you want to modify (for example, 1/8), click OpenEdit, and then set the parameters.

Enabling and Disabling Network Interfaces

Updated: 2013-08-29

By default, the network interfaces are enabled. You must disable any network interface that is not connected to the network, so that it cannot send or receive packets. Disabling a network interface that is connected to the network in a high availability setup can cause failover.

For more information about high availability, see "High Availability."

To enable or disable a network interface by using the command line interface

At the command prompt, type one of the following pairs of commands to enable or disable an interface and verify the setting:

- enable interface <interface_num>
- show interface <interface_num>
- disable interface <interface_num>
- show interface <interface_num>

Example

```
> enable interface 1/8
 Done
> show interface 1/8
      Interface 1/8 (Gig Ethernet 10/100/1000 MBits) #2
      flags=0x4004000 <ENABLED, DOWN, BOUND to LA/1, down, autoneg, 802.1q>
      MTU=1514, MAC=00:d0:68:15:fd:3d, downtime 906h58m40s
      Requested: media UTP, speed AUTO, duplex FULL, fctl OFF, throughput 0
      RX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
      TX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
      NIC: InDisc(0) OutDisc(0) Fctls(0) Stalls(0) Hangs(0) Muted(0)
      Bandwidth thresholds are not set.
Done
```

To enable or disable a network interface by using the configuration utility

1. Navigate to System > Network > Interfaces.
2. Select the network interface and, in the Action list, select Enable or Disable.

Resetting Network Interfaces

Updated: 2013-09-30

Network interface settings control properties such as duplex and speed. To renegotiate the settings of a network interface, you must reset it.

To reset a network interface by using the command line interface

At the command prompt, type the following commands to reset an interface and verify the setting:

- reset interface <interface_num>
- show interface <interface_num>

Example

```
> reset interface 1/8
 Done
```

To reset a network interface by using the configuration utility

1. Navigate to System > Network > Interfaces.
2. Select the network interface and, in the Action list, select Reset Interface.

Monitoring a Network Interface

Updated: 2013-08-29

You can display network interface statistics to monitor parameters such as packets sent and packets received, throughput, Link Aggregate Control Protocol (LACP) data units, and errors, and use the information to check the health of the network interface. You can clear the statistics of a network interface to monitor its statistics from the time the statistics are cleared.

To display the statistics of the network interfaces by using the command line interface

At the command prompt, type:

stat interface <interface_num>

To display the statistics of an Interface by using the configuration utility

Navigate to System > Network > Interfaces, select the network interface, and click StatisticsInterface Statistics.

To clear a network interface's statistics by using the command line interface

At the command prompt, type:

clear interface <interface_num>

Example

```
> clear interface 1/8
 Done
```

To clear a network interface's statistics by using the configuration utility

1. Navigate to System > Network > Interfaces.
2. Select the network interface and, in the Action list, select Clear Statistics.

Configuring Forwarding Session Rules

Jun 11, 2014

By default, the NetScaler appliance does not create session entries for traffic that it only forwards (L3 mode). For a case in which a client request that the appliance forwards to a server results in a response that has to return by the same path, you can create a forwarding-session rule. A forwarding-session rule creates forwarding-session entries for traffic that originates from or is destined for a particular network and is forwarded by the NetScaler. You can create forwarding session rules for IPv4 traffic as well as IPv6 traffic.

When configuring an IPv4 forwarding-session rule, you can specify either an IPv4 network address or an extended ACL as the condition for identifying IPv4 traffic for which to create a forwarding-session entry:

- **Network address**. When you specify an IPv4 network address, the appliance creates forwarding sessions for IPv4 traffic whose source or destination matches the network address.
- **Extended ACL rule**. When you specify an extended ACL rule, the appliance creates forwarding sessions for IPv4 traffic that matches the conditions specified in the extended ACL rule.

When configuring an IPv6 forwarding-session rule, you can specify either an IPv6 prefix or an ACL6 as the condition for identifying IPv6 traffic for which to create a forwarding-session entry:

- **IPv6 prefix**. When you specify an IPv6 prefix, the appliance creates forwarding sessions for IPv6 traffic whose source or destination matches the IPv6 prefix.
- **ACL6 rule**. When you specify an ACL6 rule, the appliance creates forwarding sessions for IPv6 traffic that matches the conditions specified in the ACL6 rule.

To create an IPv4 forwarding session rule by using the command line interface

At the command prompt, type the following commands to create a forwarding-session rule and verify the configuration:

- add forwardingSession <name> [<network> <netmask>] | [-aclname <string>] -connfailover (ENABLED | DISABLED)
- show forwardingSession

Example

A network address as the condition:

```
> add forwardingSession fs-nw-1 10.102.105.51 255.255.255.255
 Done
```

An ACL as the condition:

```
> add forwardingSession fs-acl-1 acl1
 Done
```

To configure an IPv4 forwarding session rule by using the configuration utility

Navigate to System > Network > Forwarding Sessions, add a new IPv4 forwarding session, or edit an existing forwarding session.

To create an IPv6 forwarding session rule by using the command line interface

At the command prompt, type the following commands to create a forwarding-session rule and verify the configuration:

- add forwardingSession <name> [<IPv6 prefix>] | [-acl6name <string>]

- show forwardingSession

Example

An IPv6 prefix as the condition:

```
> add forwardingSession fsv6-pfx-1 3ffe::/64
 Done
```

An ACL6 rule as the condition:

```
> add forwardingSession fsv6-acl6-1 –acl6name ACL6-FS
 Done
```

To configure an IPv6 forwarding session rule by using the configuration utility

Navigate to System > Network > Forwarding Sessions, add a new IPv6 forwarding session, or edit an existing forwarding session.

Understanding VLANs

Mar 19, 2012

A NetScaler appliance supports Layer 2 port and IEEE 802.1q tagged VLANs. VLAN configurations are useful when you need to restrict traffic to certain groups of stations. You can configure a network interface as a part of multiple VLANs by using IEEE 802.1q tagging.

You can configure VLANs and bind them to IP subnets. The NetScaler then performs IP forwarding between these VLANs (if it is configured as the default router for the hosts on these subnets).

The NetScaler supports the following types of VLANs:

Port-Based VLANs. The membership of a port-based VLAN is defined by a set of network interfaces that share a common, exclusive Layer 2 broadcast domain. You can configure multiple port-based VLANs. By default, all network interfaces on the NetScaler are members of VLAN 1.

If you apply 802.1q tagging to the port, the network interface belongs to a port-based VLAN. Layer 2 traffic is bridged within a port-based VLAN, and Layer 2 broadcasts are sent to all members of the VLAN if Layer 2 mode is enabled. When you add an untagged network interface as a member of a new VLAN, it is removed from its current VLAN.

Default VLAN. By default, the network interfaces on the NetScaler are included in a single, port-based VLAN as untagged network interfaces. This VLAN is the default VLAN. It has a VLAN ID (VID) of 1. This VLAN exists permanently. It cannot be deleted, and its VID cannot be changed.

When you add a network interface to a to a different VLAN as an untagged member, the network interface is automatically removed from the default VLAN. If you unbind a network interface from its current port-based VLAN, it is added to the default VLAN again.

Tagged VLANs. 802.1q tagging (defined in the IEEE 802.1q standard) allows a networking device (such as the NetScaler) to add information to a frame at Layer 2 to identify the VLAN membership of the frame. Tagging allows network environments to have VLANs that span multiple devices. A device that receives the packet reads the tag and recognizes the VLAN to which the frame belongs. Some network devices do not support receiving both tagged and untagged packets on the same network interface—in particular, Force10 switches. In such cases, you need to contact customer support for assistance.

The network interface can be a tagged or untagged member of a VLAN. Each network interface is an untagged member of one VLAN only (its native VLAN). This network interface transmits the frames for the native VLAN as untagged frames. A network interface can be a part of more than one VLAN if the other VLANs are tagged.

When you configure tagging, be sure to match the configuration of the VLAN on both ends of the link. The port to which the NetScaler connects must be on the same VLAN as the NetScaler network interface.

Note: This VLAN configuration is neither synchronized nor propagated, therefore you must perform the configuration on each unit in an HA pair independently.

Applying Rules to Classify Frames

VLANs have two types of rules for classifying frames:

Ingress rules. Ingress rules classify each frame as belonging only to a single VLAN. When a frame is received on a network interface, the following rules are applied to classify the frame:

- If the frame is untagged, or has a tag value equal to 0, the VID of the frame is set to the port VID (PVID) of the receiving interface, which is classified as belonging to the native VLAN. (PVIDs are defined in the IEEE 802.1q standard.)
- If frame has a tag value equal to FFF, the frame is dropped.
- If the VID of the frame specifies a VLAN of which the receiving network interface is not a member, the frame is dropped. For example, if a packet is sent from a subnet associated with VLAN ID 12 to a subnet associated with VLAN ID 10, the packet is dropped. If an untagged packet with VID 9 is sent from the subnet associated with VLAN ID 10 to a network interface PVID 9, the packet is dropped.

Egress Rules. The following egress rules are applied:
- If the VID of the frame specifies a VLAN of which the transmission network interface is not a member, the frame is discarded.
- During the learning process (defined by the IEEE 802.1q standard), the Src MAC and VID are used to update the bridge lookup table of the NetScaler.
- A frame is discarded if its VID specifies a VLAN that does not have any members. (You define members by binding network interfaces to a VLAN.)

VLANs and Packet Forwarding on the NetScaler

The forwarding process on the NetScaler appliance is similar to that on any standard switch. However, the NetScaler performs forwarding only when Layer 2 mode is on. The key features of the forwarding process are:
- Topology restrictions are enforced. Enforcement involves selecting each network interface in the VLAN as a transmission port (depending on the state of the network interface), bridging restrictions (do not forward on the receiving network interface), and MTU restrictions.
- Frames are filtered on the basis of information in the bridge table lookup in the forwarding database (FDB) table of the NetScaler. The bridge table lookup is based on the destination MAC and the VID. Packets addressed to the MAC address of the NetScaler are processed at the upper layers.
- All broadcast and multicast frames are forwarded to each network interface that is a member of the VLAN, but forwarding occurs only if L2 mode is enabled. If L2 mode is disabled, the broadcast and multicast packets are dropped. This is also true for MAC addresses that are not currently in the bridging table.
- A VLAN entry has a list of member network interfaces that are part of its untagged member set. When forwarding frames to these network interfaces, a tag is not inserted in the frame.
- If the network interface is a tagged member of this VLAN, the tag is inserted in the frame when the frame is forwarded.

When a user sends any broadcast or multicast packets without the VLAN being identified, that is, during duplicate address detection (DAD) for NSIP or ND6 for the next hop of the route, the packet is sent out on all the network interfaces, with appropriate tagging based on either the Ingress and Egress rules. ND6 usually identifies a VLAN, and a data packet is sent on this VLAN only. Port-based VLANs are common to IPv4 and IPv6. For IPv6, the NetScaler supports prefix-based VLANs.

Configuring a VLAN

May 26, 2015

You can implement VLANs in the following environments:

- Single subnet
- Multiple subnets
- Single LAN
- VLANs (no tagging)
- VLANs (802.1q tagging)

If you configure VLANs that have only untagged network interfaces as their members, the total number of possible VLANs is limited to the number of network interfaces available in the NetScaler. If more IP subnets are required with a VLAN configuration, 802.1q tagging must be used.

When you bind a network interface to a VLAN, the network interface is removed from the default VLAN. If the network interfaces need to be a part of more than one VLAN, you can bind the network interfaces to the VLANs as tagged members.

You can configure the NetScaler to forward traffic between VLANs at Layer 3. In this case, a VLAN is associated with a single IP subnet. The hosts in a VLAN that belong to a single subnet use the same subnet mask and one or more default gateways connected to that subnet. Configuring Layer 3 for a VLAN is optional. Layer 3 is used for IP forwarding (inter-VLAN routing). Each VLAN has a unique IP address and subnet mask that define an IP subnet for the VLAN. In an HA configuration, this IP address is shared with the other NetScaler appliances. The NetScaler forwards packets between configured IP subnets (VLANs).

When you configure the NetScaler, you must not create overlapping IP subnets. Doing so impedes Layer 3 functionality.

Each VLAN is a unique Layer 2 broadcast domain. Two VLANs, each bound to separate IP subnets, cannot be combined into a single broadcast domain. Forwarding traffic between two VLANs requires a Layer 3 forwarding (routing) device, such as the NetScaler appliance.

Configuring VLANs in an HA Setup

VLAN configuration for a high-availability setup requires that the NetScaler appliances have the same hardware configuration, and the VLANs configured on them must be mirror images.

The correct VLAN configuration is implemented automatically when the configuration is synchronized between the NetScaler appliances. The result is identical actions on all the appliances. For example, adding network interface 0/1 to VLAN2 adds this network interface to VLAN 2 on all the appliances participating in the high-availability setup.

Note: If you use network-interface-specific commands in an HA setup, the configurations you create are not propagated to the other NetScaler appliance. You must perform these commands on each appliance in an HA pair to ensure that the configuration of the two appliances in the HA pair remains synchronized.

Creating or Modifying a VLAN

Updated: 2013-08-29

To configure a VLAN, you create a VLAN entity, and then bind network interfaces and IP addresses to the VLAN. If you remove a VLAN, its member interfaces are added to the default VLAN.

To create a VLAN by using the command line interface

At the command prompt, type:

add vlan <id> [-aliasName <string>] [-ipv6DynamicRouting (ENABLED | DISABLED)]
Example

```
> add vlan 2 –aliasName “Network A”  Done
```

To bind an interface to a VLAN by using the command line interface

At the command prompt, type:

bind vlan <id> -ifnum <slot/port>
Example

```
> bind vlan 2 -ifnum 1/8  Done
```

To bind an IP address to a VLAN by using the command line interface

At the command prompt, type:

bind vlan <id> -IPAddress <IPAddress> <netMask>
Example

```
> bind vlan 2 -IPAddress 10.102.29.54 255.255.255.0  Done
```

To remove a VLAN by using the command line interface

At the command prompt, type:

rm vlan <id>

To configure a VLAN by using the configuration utility

1. Navigate to System > Network > VLANs, add a new VLAN, or edit an existing VLAN.
2. To bind an IP address to a VLAN, under IP Bindings, select the Active option corresponding to the IP address that you want to bind to the VLAN (for example, 10.102.29.54). The Type column displays the IP address type (such as mapped IP, virtual IP, or subnet IP) for each IP address in the IP Address column.
3. To bind a network interface to a VLAN, under Interface Bindings, select the Active option corresponding to the interface that you want to bind to the VLAN.

Monitoring VLANS

Updated: 2013-08-29

You can display VLAN statistics such as packets received, bytes received, packets sent, and bytes sent, and use the information to identify anomalies and or debug a VLAN.

To view the statistics of a VLAN by using the command line interface

At the command prompt, type:

stat vlan <vlanID>
Example


```
stat vlan 2
```

To view the statistics of a VLAN by using the configuration utility

1. Navigate to System > Network > VLANs.
2. Select the VLAN, and click Statistics.

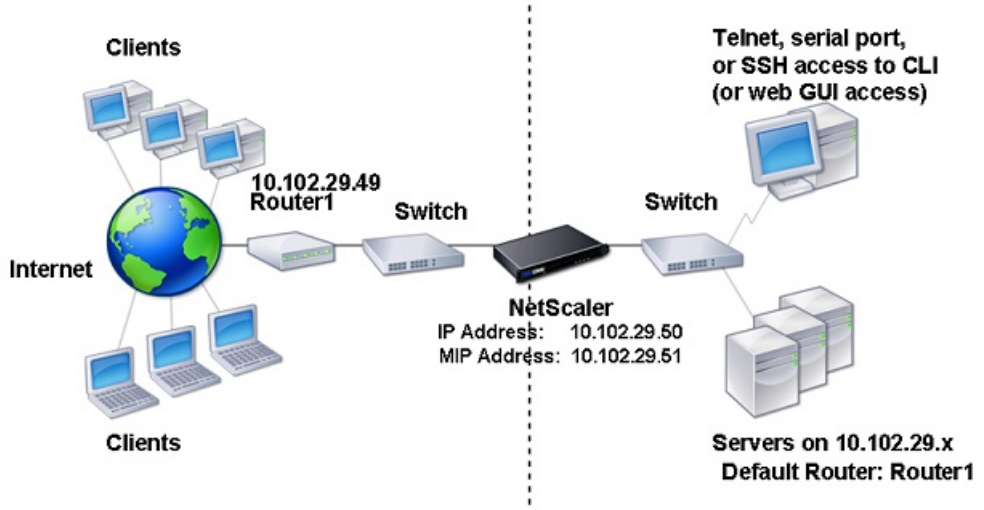
Configuring VLANs on a Single Subnet

Aug 29, 2013

Before configuring a VLAN on a single subnet, make sure that Layer 2 Mode is enabled.

The following figure shows a single subnet environment

Figure 1. VLAN on a Single Subnet



In the above figure:

1. The default router for the NetScaler and the servers is Router 1.
2. Layer 2 mode must be enabled on the NetScaler for the NetScaler to have direct access to the servers.
3. For this subnet, a virtual server can be configured for load balancing on the NetScaler appliance.

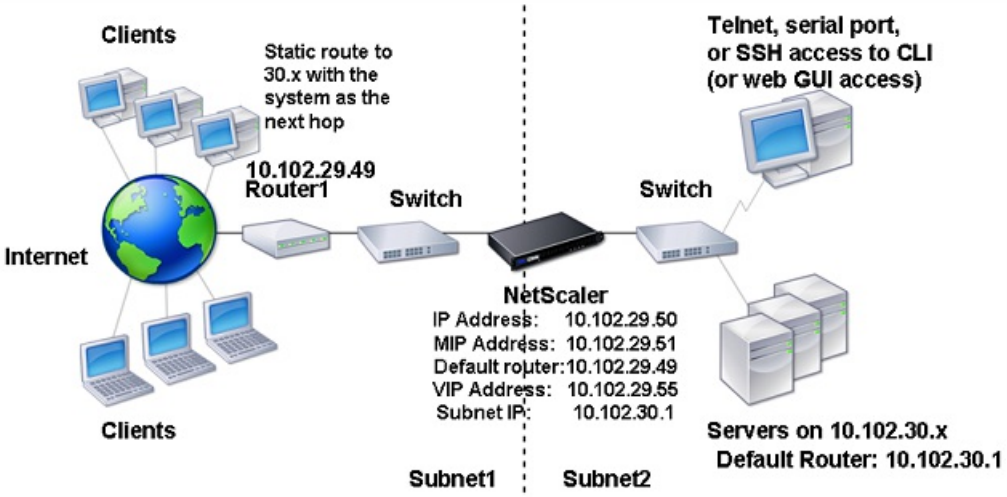
To configure a VLAN on a single subnet, follow the procedures described in "Creating or Modifying a VLAN" in Configuring a VLAN. VLAN configuration parameters are not required, because the network interfaces are members of this VLAN.

Configuring VLANs on Multiple Subnets

Aug 29, 2013

To configure a single VLAN across multiple subnets, you must add a VIP for the VLAN and configure the routing appropriately. The following figure shows a single VLAN configured across multiple subnets.

Figure 1. Multiple Subnets in a Single VLAN



To configure a single VLAN across multiple subnets, perform the following tasks:

1. Disable Layer 2 mode. For the procedure to disable Layer 2 mode, see "Enabling and Disabling Layer 2 Mode."
2. Add a VIP.

   For the procedure to add a VIP, see "Configuring and Managing Virtual IP Addresses (VIPs)."
3. Configure RNAT ID.

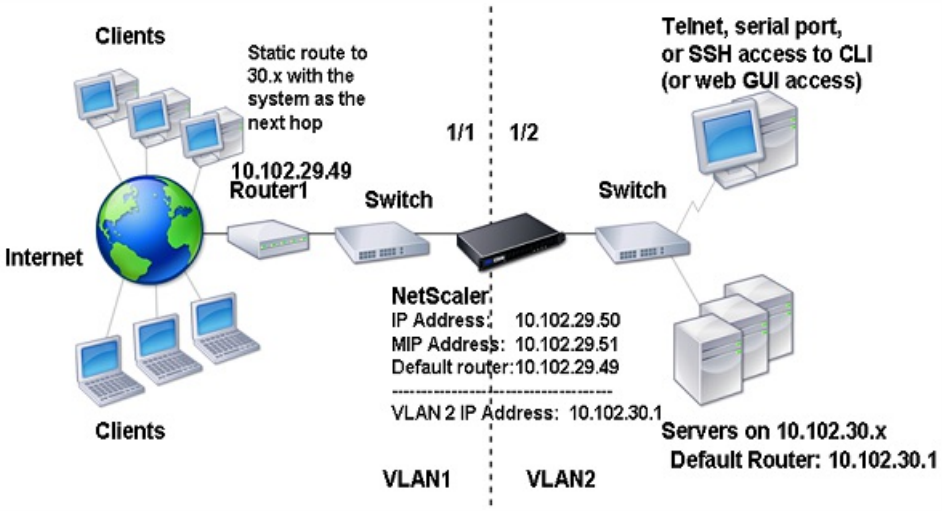
   For the procedure to configure the RNAT ID, see "Configuring RNAT."

Configuring Multiple Untagged VLANs across Multiple Subnets

Aug 29, 2013

In environments with multiple untagged VLANs across multiple subnets, a VLAN is configured for each IP subnet. A network interface is bound to one VLAN only. The following figure shows this configuration.

Figure 1. Multiple Subnets with VLANs - No Tagging



To implement the configuration shown in the above figure, perform the following tasks:

1. Add VLAN 2.

   For the procedure to create a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

2. Bind the 1/2 network interface of the NetScaler to VLAN 2 as an untagged network interface.

   For the procedure to bind a network interface to a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

3. Bind the IP address and subnet mask to VLAN 2.

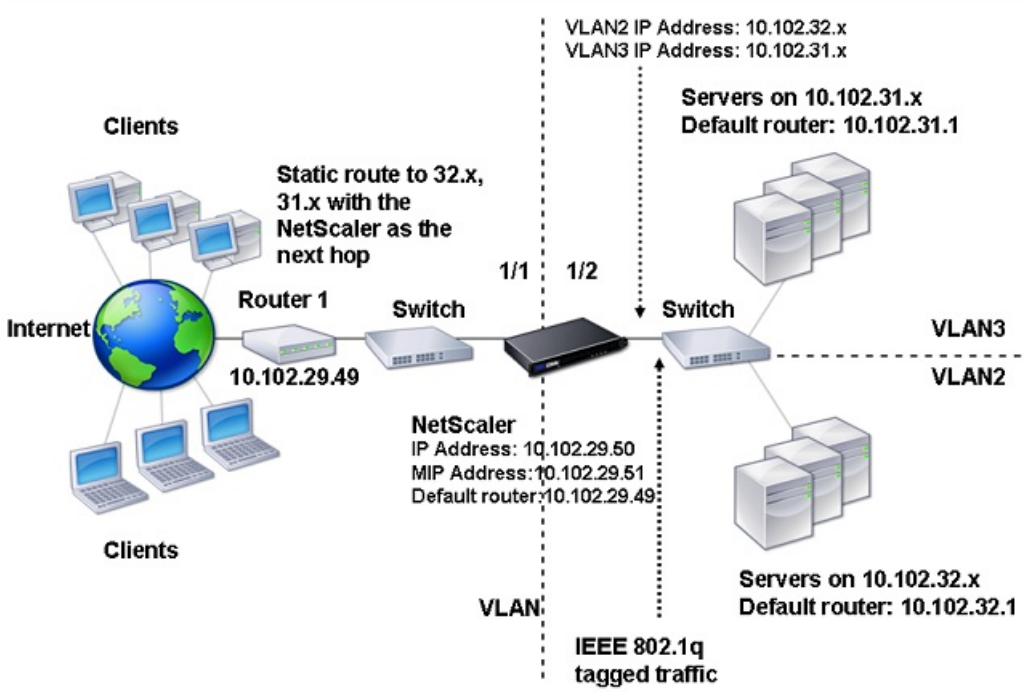
   For the procedure to bind a network interface to a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

Configuring Multiple VLANs with 802.1q Tagging

Aug 29, 2013

For multiple VLANs with 802.1q tagging, each VLAN is configured with a different IP subnet. Each network interface is in one VLAN. One of the VLANs is set up as tagged. The following figure shows this configuration.

Figure 1. Multiple VLANs with IEEE 802.1q Tagging



To implement the configuration shown in the above figure, perform the following tasks:

1. Add VLAN 2.

   For the procedure to create a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

2. Bind the 1/2 network interface of the NetScaler to VLAN 2 as an untagged network interface.

   For the procedure to bind a network interface to a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

3. Bind the IP address and netmask to VLAN 2.

   For the procedure to bind an IP address to a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

4. Add VLAN 3.

   For the procedure to create a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

5. Bind the 1/2 network interface of the NetScaler to VLAN 3 as a tagged network interface.

   For the procedure to bind a network interface to a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

   For the procedure to bind a tagged network interface, see "Creating or Modifying a VLAN" in Configuring a VLAN.

6. Bind the IP address and netmask to VLAN 3.

For the procedure to bind an IP address to a VLAN, see "Creating or Modifying a VLAN" in Configuring a VLAN.

Configuring NSVLAN

Jul 05, 2013

NSVLAN is a VLAN to which the NetScaler management IP (NSIP) address's subnet is bound. The NSIP subnet is available only on interfaces that are associated with NSVLAN. By default, NSVLAN is VLAN1, but you can designate a different VLAN as NSVLAN. If you do so, you must reboot the NetScaler appliance for the change to take effect. After the reboot, NSIP subnet traffic is restricted to the new NSVLAN.

The traffic from the NetScaler IP subnet can be tagged (802.1q) with the VLAN ID specified for NSVLAN. You must configure the attached switch interface to tag and allow this same VLAN ID on the connected interface.

If you remove your NSVLAN configuration, the NSIP subnet is automatically bound to VLAN1, restoring the default NSVLAN.

To configure NSVLAN by using the command line interface

At the command prompt, type:

- set ns config -nsvlan <positive_integer> -ifnum <interface_name> ... [-tagged (YES | NO)]
- show ns config

Note: The configuration takes effect after the NetScaler appliance is rebooted.

Example

```
> set ns config -nsvlan 300 -ifnum 1/1 1/2 1/3 -tagged NO
 Done

> save config
 Done
```

To restore the default NSVLAN configuration by using the command line interface

At the command prompt, type:

- unset ns config -nsvlan
- show ns config

Example

```
> unset ns config -nsvlan
 Done
```

To configure NSVLAN by using the configuration utility

Navigate to System > Settings, in the Settings group, click Change NSVLAN Settings.

Configuring Bridge Groups

Aug 29, 2013

Typically, when you want to merge two or more VLANs into a single domain, you change the VLAN configuration on all the devices in the separate domains. This can be a tedious task. To more easily merge multiple VLANs into a single broadcast domain, you can use bridge groups.

The bridge groups feature works the same way as a VLAN. Multiple VLANS can be bound to a single bridge group, and all VLANs bound to same bridge group form a single broadcast domain. You can bind only Layer 2 VLANs to a bridge group. For Layer 3 functionality, you must assign an IP address to a bridge group.

In Layer 2 mode, a broadcast packet received on an interface belonging to a particular VLAN is bridged to other VLANs that belong to the same bridge group. In the case of a unicast packet, the NetScaler appliance searches its bridge table for the learned MAC addresses of all the VLANs belonging to same bridge group.

In Layer 3 forwarding mode, an IP subnet is bound to a bridge group. The NetScaler accepts incoming packets belonging to the bound subnet and forwards the packets only on VLANs that are bound to the bridge group.

IPv6 routing can be enabled on a configured bridge group.

To add a bridge group and bind VLANs by using the command line interface

To add a bridge group and bind VLANs and verify the configuration, type the following commands:

- add bridgegroup <id> [-ipv6DynamicRouting (ENABLED | DISABLED)]
- show bridgegroup <id>
- bind bridgegroup <id> -vlan <positive_integer>
- show bridgegroup <id>

Example

```
> add bridgegroup 12
 Done
```

To remove a bridge group by using the command line interface

At the command prompt, type:

rm bridgegroup <id>

Example

```
 rm bridgegroup 12
```

To configure a bridge group by using the configuration utility

Navigate to System > Network > Bridge Groups, add a new bridge group, or edit an existing bridge group.

Configuring VMACs

Aug 29, 2013

The primary and secondary nodes in a high availability (HA) setup share the Virtual MAC address (VMAC) floating entity. The primary node owns the floating IP addresses (such as MIP, SNIP, and VIP) and responds to ARP requests for these IP addresses with its own MAC address. Therefore, the ARP table of an external device, such as an upstream router, is updated with the floating IP address and the MAC address of the primary node.

When a failover occurs, the secondary node takes over as the new primary node. The former secondary node uses Gratuitous ARP (GARP) to advertise the floating IP addresses that it had learned from the old primary node. The MAC address that the new primary node advertises is the MAC address of its own network interface. Some devices (a few routers) do not accept these GARP messages. Therefore, these external devices retain the IP address-to-MAC address mapping that the old primary node had advertised. This can result in a GSLB site going down.

Therefore, you must configure a VMAC on both nodes of an HA pair. This means that both nodes have identical MAC addresses. When a failover occurs, the MAC address of the secondary node remains unchanged, and the ARP tables on the external devices do not need to be updated.

For the procedures to configure a VMAC, see "High Availability."

Configuring Link Aggregation

Sep 06, 2013

Link aggregation combines data coming from multiple ports into a single high-speed link. Configuring link aggregation increases the capacity and availability of the communication channel between the NetScaler appliance and other connected devices. An aggregated link is also referred to as a "channel." You can configure the channels manually, or you can use Link Aggregation Control Protocol (LACP). You cannot apply LACP to a manually configured channel, nor can you manually configure a channel created by LACP.

When a network interface is bound to a channel, the channel parameters have precedence over the network interface parameters. (That is, the network interface parameters are ignored.) A network interface can be bound only to one channel.

When a network interface is bound to a channel, it drops its VLAN configuration. When network interfaces are bound to a channel, either manually or by LACP, they are removed from the VLANs that they originally belonged to and added to the default VLAN. However, you can bind the channel back to the old VLAN, or to a new one. For example, if you bind the network interfaces 1/2 and 1/3 to a VLAN with ID 2, and then bind them to a channel LA/1, the network interfaces are moved to the default VLAN, but you can bind them back to VLAN 2.

This section includes the following details:

- Configuring Link Aggregation Manually
- Configuring Link Aggregation by Using the Link Aggregation Control Protocol
- Configuring Link Redundancy using LACP channels

Configuring Link Aggregation Manually

Updated: 2013-08-29

When you create a link aggregation channel, its state is DOWN until you bind an active interface to it. You can modify a channel at any time. You can remove channels, or you can enable/disable them.

To create a link aggregation channel by using the command line interface

At the command prompt, type:

- add channel <id> [-ifnum <interfaceName> ...] [-state (ENABLED | DISABLED)] [-speed <speed>] [-flowControl <flowControl>] [-haMonitor (ON | OFF)][-tagall (ON | OFF)] [-ifAlias <string>] [-throughput <positive_integer>] [-bandwidthHigh <positive_integer> [-bandwidthNormal <positive_integer>]]
- show channel

Example

```
add channel LA/1 -ifnum 1/8
show channels
```

To bind an interface to or unbind an interface from an existing link aggregation channel by using the command line interface

At the command prompt, type one of the following commands:

- bind channel <id> <interfaceName>
- unbind channel <id> <interfaceName>

Example

```
bind channel LA/1 1/8
```

To modify a link aggregation channel by using the command line interface

At the command prompt, type the set channel command, the channel ID, and the parameters to be changed, with their new values.

To configure a link aggregation channel by using the configuration utility

Navigate to System > Network > Channels, add a new channel, or edit an existing channel.

To remove a link aggregation channel by using the command line interface

Important: When a channel is removed, the network interfaces bound to it induce network loops that decrease network performance. You must disable the network interfaces before you remove the channel.
At the command prompt, type:

rm channel <id>
Example

```
> rm channel LA/1
 Done
```

To remove a link aggregation channel by using the configuration utility

Important: When a channel is removed, the network interfaces bound to it induce network loops that decrease network performance. You must disable the network interfaces before you remove the channel.
Navigate to System > Network > Channels, select the channel that you want to remove and click Remove.

Configuring Link Aggregation by Using the Link Aggregation Control Protocol

The Link Aggregation Control Protocol (LACP) enables network devices to exchange link aggregation information by exchanging LACP Data Units (LACPDUs). Therefore, you cannot enable LACP on network interfaces that are members of a channel that you created manually.

When using LACP to configure link aggregation, you use different commands and parameters for modifying link aggregation channels than you do for creating link aggregation channels. To remove a channel, you must disable LACP on all interfaces that are part of the channel.

Note: In an High Availability configuration, LACP configurations are neither propagated nor synchronized.

Configuring the LACP System Priority

Updated: 2013-10-01

The LACP system priority determines which peer device of an LACP LA channel can have control over the LA channel. This number is globally applied to all LACP channels on the appliance. The lower the value, the higher the priority.

To configure the LACP system priority by using the command line interface

At the command prompt, type the following commands to set the priority for a standalone appliance and verify the configuration:

- set lacp -sysPriority <positive_integer>
- show lacp

Example:

```
set lacp -sysPriority 50
```

To set the priority for a specific cluster node, log on to the cluster IP address and at the command prompt, type the following commands:

- set lacp -sysPriority <positive_integer> -ownerNode <positive_integer>
- show lacp

Example:

```
set lacp -sysPriority 50 -ownerNode 2
```

To configure the LACP system priority by using the configuration utility

1. Navigate to System > Network > Interfaces and, in the Action list, select Set LACP.
2. Specify the system priority and the owner node (applicable only for a cluster setup).

Creating Link Aggregation Channels

Updated: 2013-08-29

For creating a link aggregation channel by using LACP, you need to enable LACP and specify the same LACP key on each interface that you want to be the part of the channel. For example, if you enable LACP and set the LACP Key to 3 on interfaces 1/1 and 1/2, a link aggregation channel LA/3 is created and interfaces 1/1 and 1/2 are automatically bound to it.

Note: When enabling LACP on a network interface, you must specify the LACP Key.
By default, LACP is disabled on all network interfaces.

To create an LACP channel by using the command line interface

At the command prompt, type:

- set interface <id> [-lacpMode <lacpMode>] [-lacpKey<positive_integer>] [-lacpPriority <positive_integer>] [-lacpTimeout (LONG | SHORT)]
- show interface [<id>]

To create an LACP channel by using the configuration utility

Navigate to System > Network > Interfaces, open the network interface, and set the parameters.

Modifying Link aggregation Channels

Updated: 2013-08-29

After you have created an LACP channel by specifying interfaces, you can modify properties of the channel.

To modify an LACP channel using the command line interface

At the command prompt, type:

- set channel <id> [-ifnum <interfaceName> ...] [-state (ENABLED | DISABLED)] [-speed <speed>] [-flowControl

<flowControl>] [-haMonitor (ON | OFF)] [-ifAlias <string>] [-throughput <positive_integer>] [-tagall (ON | OFF)] [-bandwidthHigh <positive_integer> [-bandwidthNormal <positive_integer>]]

- show channel

Example

```
> set channel LA/3 -state ENABLED -speed 10000
 Done
```

To modify an LACP channel by using the configuration utility

Navigate to System > Network > Channels, and modify an existing LACP channel.

Removing a Link Aggregation Channel

Updated: 2013-08-29

To remove a link aggregation channel that was created by using LACP, you need to disable LACP on all the interfaces that are part of the channel.

To remove an LACP channel by using the command line interface

At the command prompt, type:

- set interface <id> -lacpMode Disable
- show interface [<id>]

To remove an LACP channel by using the configuration utility

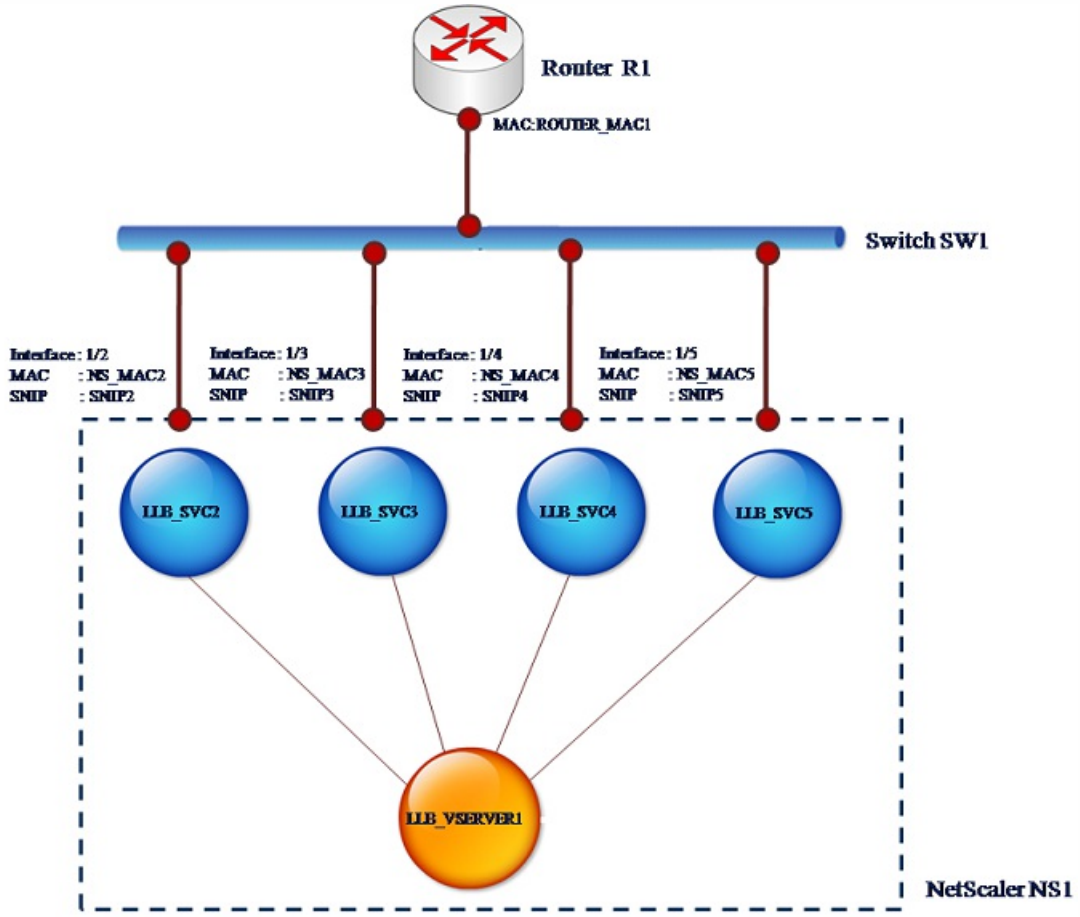
Navigate to System > Network > Interfaces, open the network interface, and clear the Enable LACP option.

Binding an SNIP address to an Interface

Aug 29, 2013

You can now bind a NetScaler owned SNIP address to an interface without using Layer 3 VLANs. Any packets related to the SNIP address will go only through the bound interface.

This feature can be useful in a scenario where the upstream switch does not support Link Aggregation channels and you want the NetScaler appliance to load balance traffic, originated from a server, across the four links to the upstream switch as shown in the following illustration.



The following tables describe the example settings for the scenario:

Entity	Name	Value
SNIP addresses on NS1	SNIP2 (for reference purpose only)	10.10.10.2
	SNIP3 (for reference purpose only)	10.10.10.3
	SNIP4 (for reference purpose only)	10.10.10.4

Entity	Name	Value
	SNIP5 (for reference purpose only)	10.10.10.5
LLB virtual server on NS1	LLB_VSERVER1	-
Transparent monitor on NS1	TRANS_MON	-
LLB services on NS1	LLB_SVC2	10.10.10.240
	LLB_SVC3	10.10.10.120
	LLB_SVC4	10.10.10.60
	LLB_SVC5	10.10.10.30
MAC address of interface 1/2 on NS1	NS_MAC_2 (for reference purpose only)	00:e0:ed:0f:bc:e0
MAC address of interface 1/3 on NS1	NS_MAC_3 (for reference purpose only)	00:e0:ed:0f:bc:df
MAC address of interface 1/4 on NS1	NS_MAC_4 (for reference purpose only)	00:e0:ed:0f:bc:de
MAC address of interface 1/5 on NS1	NS_MAC_5 (for reference purpose only)	00:e0:ed:1c:89:53
IP address of Router R1	Router_IP (for reference purpose only)	10.10.10.1
MAC address of interface of R1	ROUTER_MAC1 (for reference purpose only)	00:21:a1:2d:db:cc

To configure the example settings

1. Add four different SNIPs in different subnet ranges. This is for ARP to be resolved on four different links. For more information on creating a SNIP address, see "Configuring Subnet IP Addresses (SNIPs)."
   Command Line Interface example

```
> add ns ip 10.10.10.2 255.255.255.0 -type SNIP
 Done
> add ns ip 10.10.10.3 255.255.255.128 –type SNIP
Done
> add ns ip 10.10.10.4 255.255.255.192 –type SNIP
Done
> add ns ip 10.10.10.5 255.255.255.224 –type SNIP
Done
```

2. Add four different dummy services in the added SNIP subnets. This is to ensure that the traffic is sent out with source IP as one of the four configured SNIPs. For more information on creating a service, see "Configuring Services."
   Command Line Interface example

   ```
   > add service LLB_SVC2 10.10.10.240 any *
    Done
   > add service LLB_SVC3 10.10.10.120 any *
    Done
   > add service LLB_SVC4 10.10.10.60 any *
    Done
   > add service LLB_SVC5 10.10.10.30 any *
    Done
   ```
3. Add a transparent ping monitor for monitoring the gateway. Bind the monitor to each of the configured dummy services. This is to make the state of the services as UP. For more information on creating a transparent monitor, see "Creating and Binding a Transparent Monitor."
   Command Line Interface example

   ```
   > add monitor TRANS_MON ping -destIP 10.10.10.1 -transparent YES
    Done
   > bind monitor TRANS_MON LLB_SVC2
    Done
   > bind monitor TRANS_MON LLB_SVC3
    Done
   > bind monitor TRANS_MON LLB_SVC4
    Done
   > bind monitor TRANS_MON LLB_SVC5
    Done
   ```
4. Add a link load balancing (LLB) virtual server and bind the dummy services to it. For more information on creating an LLB virtual server, see "Configuring an LLB Virtual Server and Binding a Service."
   Command Line Interface example

   ```
   > add lb vserver LLB_VSERVER1 any
    Done
   > set lb vserver LLB_VSERVER1 -lbmethod ROUNDROBIN
    Done
   > bind lb vserver LLB_VSERVER1 LLB_SVC2
    Done
   > bind lb vserver LLB_VSERVER1 LLB_SVC2
    Done
   > bind lb vserver LLB_VSERVER1 LLB_SVC2
    Done
   > bind lb vserver LLB_VSERVER1 LLB_SVC2
    Done
   ```
5. Add the LLB virtual server as the default LLB route. For more information on creating an LLB route see "Configuring an

LLB Route."

Command Line Interface example

```
> add lb route 0.0.0.0 0.0.0.0  LLB_VSERVER1
 Done
```

6. Add an ARP entry for each of the dummy services with the MAC address of the gateway. This way the gateway is reachable through these dummy services. For more information on adding an ARP entry, see "Configuring Static ARP."

Command Line Interface example

```
> add arp -ipaddress 10.10.10.240 -mac 00:21:a1:2d:db:cc -ifnum 1/2
 Done
> add arp -ipaddress 10.10.10.120 -mac 00:21:a1:2d:db:cc -ifnum 1/3
 Done
> add arp -ipaddress 10.10.10.60 -mac 00:21:a1:2d:db:cc -ifnum 1/4
 Done
> add arp -ipaddress 10.10.10.30 -mac 00:21:a1:2d:db:cc -ifnum 1/5
 Done
```

7. Bind a specific interface to an SNIP by adding an ARP entry for each of these SNIPs. This is to ensure that the response traffic will reach the same interface through which the request went out. For more information on adding an ARP entry, see "Configuring Static ARP."

Command Line Interface example

```
> add arp -ipAddress 10.10.10.2 -mac 00:e0:ed:0f:bc:e0 -ifnum 1/2
 Done
> add arp -ipAddress 10.10.10.3 -mac 00:e0:ed:0f:bc:df -ifnum 1/3
 Done
> add arp -ipAddress 10.10.10.4 -mac 00:e0:ed:0f:bc:de -ifnum 1/4
 Done
> add arp -ipAddress 10.10.10.5 -mac 00:e0:ed:1c:89:53 -ifnum 1/5
 Done
```

Monitoring the Bridge Table and Changing the Aging time

Aug 29, 2013

NetScaler appliance bridges frames on the basis of bridge table lookup of the destination MAC address and the VLAN ID. However, the appliance performs forwarding only when Layer 2 mode is enabled.

The bridge table is dynamically generated, but you can display it, modify the aging time for the bridge table, and view bridging statistics.

All the MAC entries in the bridge table are updated with the aging time.

To change the aging time by using the command line interface

At the command prompt, type:

- set bridgetable -bridgeAge <positive_integer>
- show bridgetable

Example

```
> set bridgetable -bridgeage 70
 Done
```

To change the aging time by using the configuration utility

1. Navigate to System > Network > Bridge Table.
2. Click Change Ageing Time, and set the Ageing Time (seconds) parameter.

To view the statistics of a bridge table by using the command line interface

At the command prompt, type:

```
stat bridge
```

To view the statistics of a bridge table by using the configuration utility

Navigate to System > Network > Bridge Table, select the MAC address, and click Statistics.

Understanding NetScaler Appliances in Active-Active Mode Using VRRP

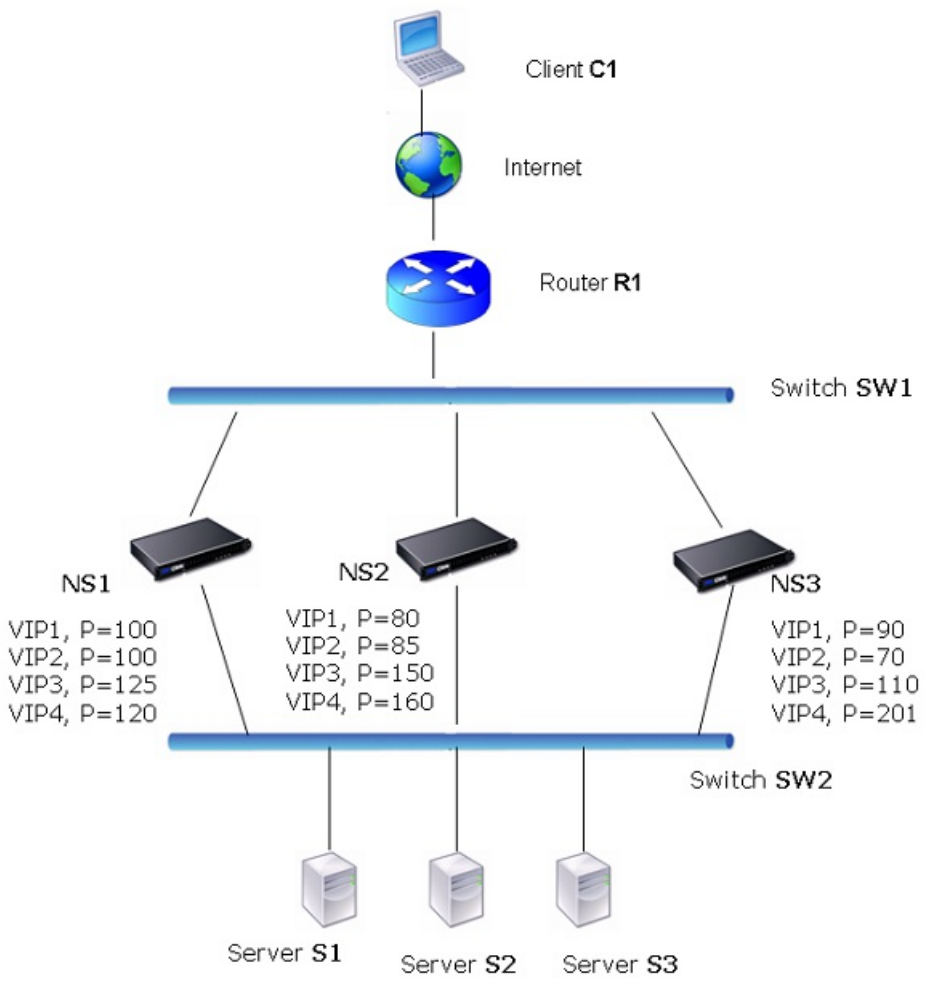
Aug 29, 2013

An active-active deployment, in addition to preventing downtime, makes efficient use of all the NetScaler appliances in the deployment. In active-active deployment mode, the same VIPs are configured on all NetScaler appliances in the configuration, but with different priorities, so that a given VIP can be active on only one appliance at a time.

Note: This feature is supported only on NetScaler nCore builds.
The active VIP is called the master VIP, and the corresponding VIPs on the other NetScaler appliances are called the backup VIPs. If a master VIP fails, the backup VIP with the highest priority takes over and becomes the master VIP. All the NetScaler appliances in an active-active deployment use the Virtual Router Redundancy Protocol (VRRP) protocol to advertise their VIPs and the corresponding priorities at regular intervals.

NetScaler appliances in active-active mode can be configured so that no NetScaler is idle. In this configuration, different sets of VIPs are active on each NetScaler. For example, in the following diagram, VIP1, VIP2, VIP3, and VIP4 are configured on appliances NS1, NS2, and NS3. Because of their priorities, VIP1 and VIP 2 are active on NS1, VIP3 is active on NS2 and VIP 4 is active on NS3. If, for example, NS1 fails, VIP1 on NS3 and VIP2 on NS2 become active.

Figure 1. An Active-Active Configuration



The NetScaler appliances in the above diagram process traffic as follows:

1. Client C1 sends a request to VIP1. The request reaches R1.
2. R1 does not have an APR entry for VIP1, so it broadcasts an ARP request for VIP1.
3. VIP1 is active in NS1, so NS1 replies with a source MAC address as the VMAC (for example VMAC1) associated with VIP1, and VIP1 as the source IP address.
4. SW1 learns the port for VIP1 from the ARP reply and updates its bridge table.
5. R1 updates the ARP entry with VMAC1 and VIP1.
6. R1 forwards the packet to the VIP1 on NS1.
7. NS1's load balancing algorithm selects server S2, and NS1 opens a connection between one of its SNIP or MIP addresses and S2.
8. S2 replies to the SNIP or MIP on the NetScaler.
9. NS1 sends S2's reply to the client. In the reply, NS1 inserts MAC address of the physical interface as the source MAC address and VIP1 as the source IP address.
10. Should NS1 fail, the NetScaler appliances use the VRRP protocol to select the VIP1 with the highest priority. In this case, VIP1 on NS3 becomes active, and the following two steps update the active-active configuration.
11. NS3 broadcasts a GARP message for VIP1. In the message, VMAC1 is the source MAC address and VIP1 is the source IP address.
12. SW1 learns the new port for VMAC1 from the GARP broadcast and updates its bridge table to send subsequent client requests for VIP1 to NS3. R1 updates its ARP table.

The priority of a VIP can be modified by health tracking. If you enable health tracking, you should make sure that preemption is also enabled, so that a VIP whose priority is lowered can be preempted by another VIP.

In some situations, traffic might reach a backup VIP. To avoid dropping such traffic, you can enable sharing, on a per-node basis, as you create an active-active configuration. Or you can enable the global send to master option. On a node on which sharing is enabled, it takes precedence over send to master.

Health Tracking

Base priority (BP-range 1-255) ordinarily determines which VIP is the master VIP, but effective priority (EP) can also affect the determination.

For example, if a VIP on NS1 has a priority of 101 and same VIP on NS2 has a priority of 99, the VIP on NS1 is active. However, if two vservers are using the VIP on NS1 and one of them goes DOWN, health tracking can reduce the EP of VIP on NS1. VRRP then makes the VIP on NS2 the active VIP.

Following are the health tracking options for modifying EP:

- **NONE.** No tracking. EP = BP
- **ALL.** If all virtual servers are UP, then EP = BP. Otherwise, EP = 0.
- **ONE.** If at least one virtual server is UP, then EP = BP. Otherwise, EP = 0.
- **PROGRESSIVE.** If ALL virtual servers are UP, then EP = BP. If ALL virtual servers are DOWN then EP = 0. Otherwise EP = BP (1 - K/N), where N is the total number of virtual servers associated with the VIP and k is the number of virtual servers that are down.

Note: If you specify a value other than NONE, preemption should be enabled, so that the backup VIP with the highest priority becomes active if the priority of the master VIP is downgraded.

Preemption

Preemption of an active VIP by another VIP that attains a higher priority is enabled by default, and normally should be enabled. In some cases, however, you may want to disable it. Preemption is a per-node setting for each VIP.

Preemption can occur in the following situations:

- An active VIP goes down and a VIP with a lower priority takes its place. If the VIP with the higher priority comes back online, it preempts the currently active VIP.
- Health tracking causes the priority of a backup VIP to become higher than that of the active VIP. The backup VIP then preempts the active VIP.

Sharing

In the event that traffic reaches a backup VIP, the traffic is dropped unless the sharing option is enabled on the backup VIP. This behavior is a per node setting for each VIP and is disabled by default.

In the figure "An Active-Active Configuration," VIP1 on NS1 is active and VIP1 VIPs on NS2 and NS3 are backups. Under certain circumstances, traffic may reach VIP1 on NS2. If Sharing is enabled on NS2, this traffic is processed instead of dropped.

Configuring Active-Active Mode

Mar 19, 2012

On each NetScaler appliance that you want to deploy in active-active mode, you must add a VMAC and bind the VMAC to a VIP. The VMAC for a given VIP must be same on each appliance. For example, if VIP 10.102.29.5, is created on the appliances, a virtual router ID must be created on each NetScaler and bound to VIP 10.102.29.5 on each NetScaler. When you bind a VMAC to a VIP, the NetScaler sends VRRP advertisements to each VLAN that is bound to that VIP. The VMAC can be shared by different VIPs configured on the same NetScaler.

This section includes the following details:

- Adding a VMAC
- Configuring Send to Master
- Configuring VRRP Communication Intervals
- Changing the Priority of a VIP Address Automatically in Active-Active Configuration (VRRP)
- An Active-Active Deployment Scenario

Adding a VMAC

Updated: 2013-08-29

To add a VMAC for an active-active configuration, you create a virtual router ID. To bind a VMAC to a VIP, you associate the VMAC's virtual router ID with the VIP.

To add a VMAC by using the command line interface

At the command prompt, type:

add vrID <value> -priority <value> -preemption (ENABLED | DISABLED) -sharing (ENABLED | DISABLED) -tracking (NONE | ONE | ALL | PROGRESSIVE)

Example

```
add vrID 125 -priority 100 -sharing ENABLED -tracking ONE
```

To add a VMAC by using the configuration utility

1. Navigate to System > Network > VMACNetwork > VMAC, on the VMAC tab, add a new VMAC, or edit an existing VMAC.
2. Set the following parameters:
   - Virtual Router ID
   - Priority
   - Tracking
   - Preemption
   - Sharing

To bind a VMAC by using the command line interface

At the command prompt, type:

set ns ip <VIP address> -vrid <value>

Example

```
set ns ip 10.102.29.5 -vrid 125
```

To bind a VMAC to a VIP by using the NetScaler configuration utility

1. Navigate to System > Network > IPs, on the IPV4s tab, open the VIP address that you want to bind to a VMAC.
2. In the Virtual Router Id drop down box, select a virtual router ID.

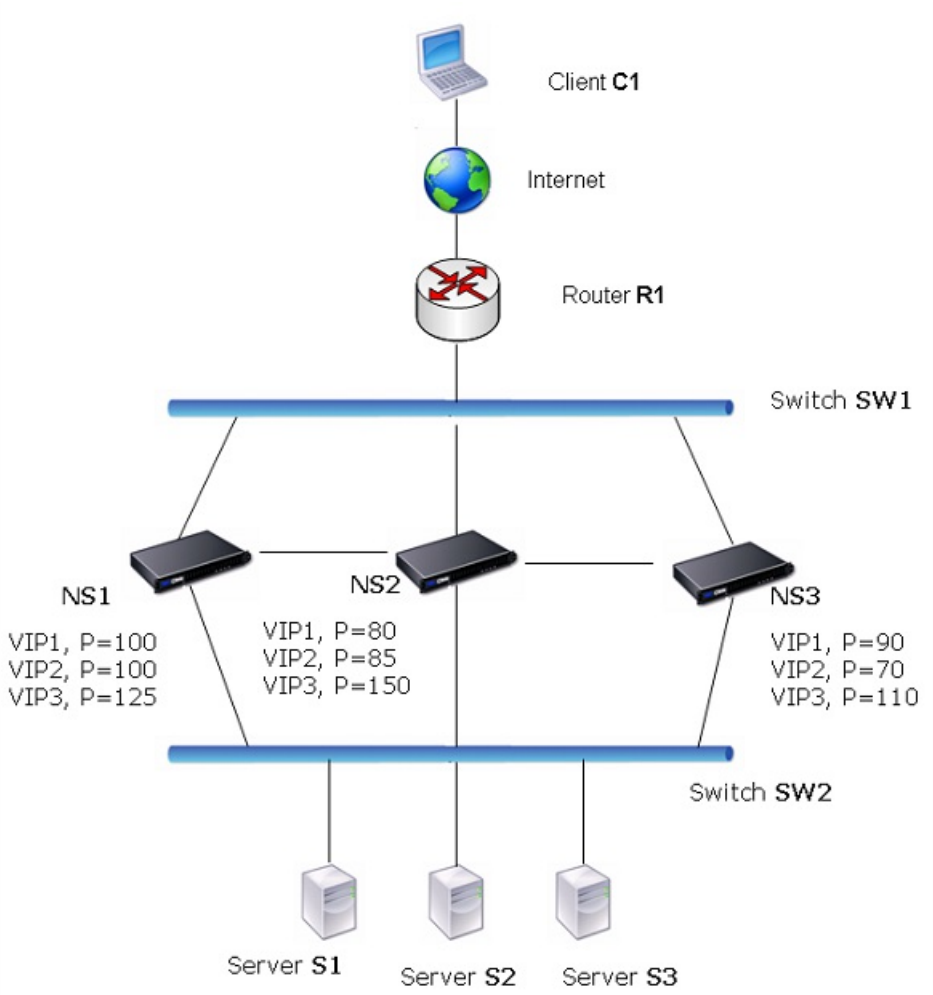
Configuring Send to Master

Updated: 2013-08-29

Usually, the traffic destined to a VIP reaches the NetScaler appliance on which the VIP is active, because an ARP request with the VIP and a VMAC on that appliance has reached the upstream router. But in some cases, such as static routes configured on the upstream router for the VIP subnet, or a topology that blocks this route, the traffic can reach a NetScaler appliance on which the VIP is in backup state. If you want this appliance to forward the data packets to the appliance on which the VIP is active, you need to enable the send to master option. This behavior is a per node setting and is disabled by default.

For example, in the following diagram, VIP1 is configured on NS1, NS2, and NS3 and is active on NS1. Under certain circumstances, traffic for VIP1 (active on NS1) may reach VIP1 on NS3. When the send to master option is enabled on NS3, NS3 forwards the traffic to NS1 through NS2 by using route entries for NS1.

Figure 1. An Active-Active Configuration with Send to Master Option Enabled



To enable send to master by using the command line interface

At the command prompt, type:

set vrIDParam -sendToMaster (ENABLED | DISABLED)

Example

```
> set vrIDParam -sendToMaster ENABLED
 Done
```

To enable send to master by using the configuration utility

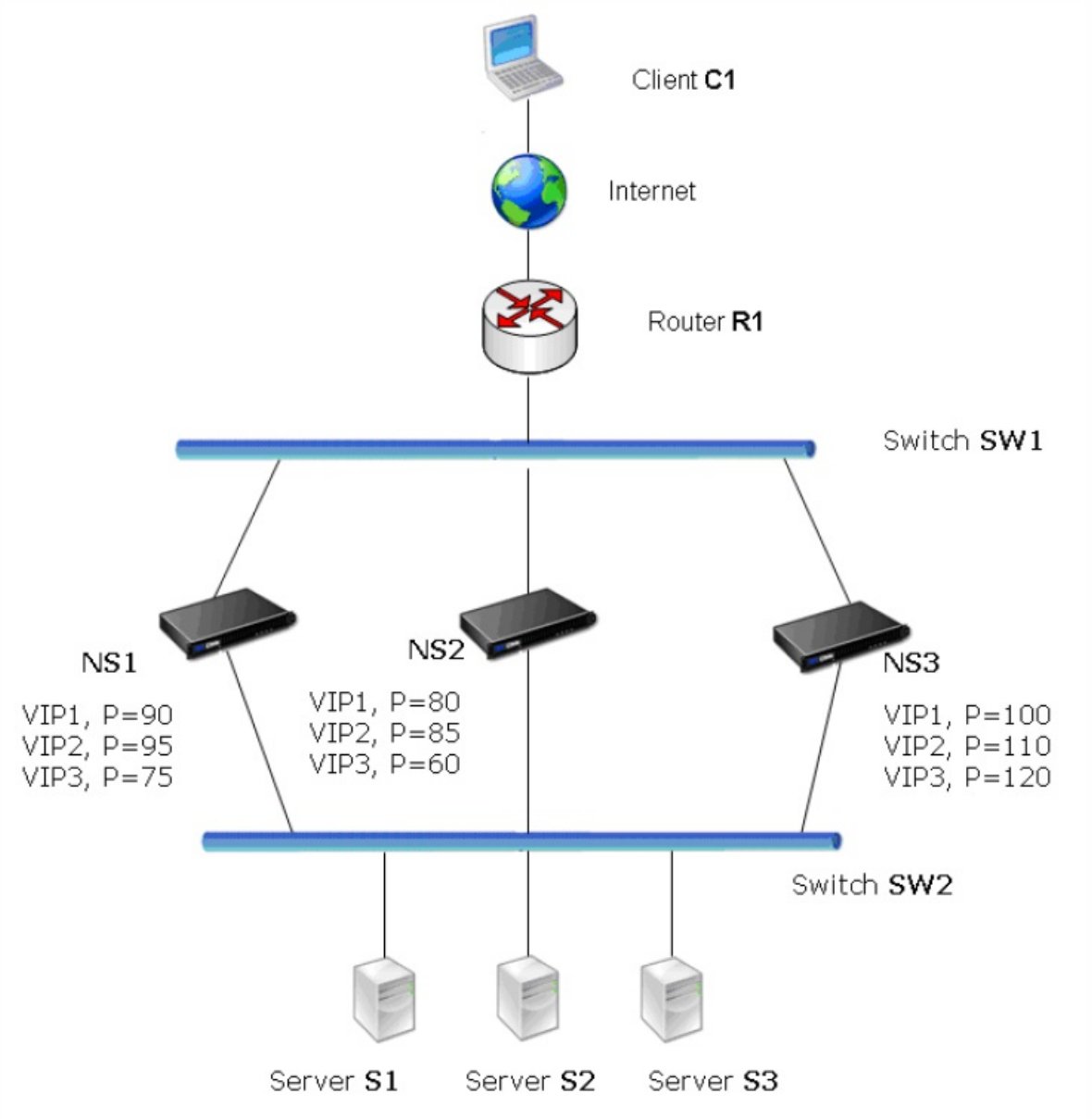
1. Navigate to System > Network, in the Settings group, click Virtual Router Parameters.
2. Select the Send to Master option.

An Active-Active Deployment Scenario

Following is an example of a possible active-active deployment scenario.

In the following diagram, VIP1, VIP 2 and VIP3 are configured on all three appliances, NS1, NS2, and NS3. Base Priorities for each VIPs are as shown in the diagram. Health tracking is disabled for each VIP. The priorities of VIPs are set so that VIP1, VIP2, and VIP3 are active on NS3. If NS3 fails, VIP1, VIP2, and VIP3 become active on NS1.

Figure 2. An Active-Active Deployment Scenario



Using the Network Visualizer

Aug 29, 2013

The Network Visualizer is a tool that you can use to view the network configuration of a NetScaler node, including the network configuration of the nodes in a high availability (HA) deployment. You can also modify the configuration of VLANs, interfaces, channels, and bridge groups, and perform HA configuration tasks.

In an HA deployment, you can both view and configure network entities on the node to which you are logged on, but you can view the details of only the network entities that are configured on the peer node. However, you can perform certain tasks, such as viewing details and statistics of the peer node and forcing a failover.

When you are logged on to a standalone appliance, you can use the Network Visualizer to do the following:

- View a consolidated graphical summary of key network components, such as VLANs, interfaces, channels, and bridge groups. You can also view the individual details of various network components.
- Modify appliance settings.
- Add, modify, and enable and disable interfaces and channels that are configured on the NetScaler appliance.
- Add and modify VLANs and bridge groups.
- Configure an HA deployment (add a node).
- View node details, node statistics, and statistics for VLANs and interfaces.
- Copy the properties of a network entity to a document or spreadsheet.

When you are logged on to an appliance in an HA deployment, you can perform the above tasks only on the appliance to which you are logged on. Following are additional tasks that you can perform in the Network Visualizer when you are logged on to one of the appliances in an HA pair:

- View the configuration details and high availability details of both nodes in an HA pair.
- Perform HA configuration tasks, such as synchronization and force failover.
- Remove the peer node from the HA configuration.
- View statistics for the peer node.
- Copy the properties of the peer node to a document or spreadsheet.

To open the Network Visualizer

1. Navigate to System > Network.
2. In Monitor Connections, click Network Visualizer.

To locate a VLAN or bridge group in the Visualizer

1. Open the Network Visualizer, and then do the following:

   - To locate a VLAN or bridge group, in the Search text field, begin typing the ID of the VLAN or the bridge group that you want to locate.
     Alternatively, begin typing the IP address of a bound subnet or the ID of a bound interface. The VLANs or bridge groups whose names match the typed characters are highlighted.

     To highlight multiple entities simultaneously, separate the IDs and IP addresses with white spaces. Entities whose IDs or IP addresses match any of the typed IDs and IP addresses are highlighted.

   - To clear the Search field, click the x adjacent to the field.

To modify the network settings of the appliance by using the Visualizer

1. Open the Network Visualizer and click the icon representing the appliance to which you are logged on.
2. In Related Tasks, click Open.

To add a channel by using the Visualizer

1. Open the Network Visualizer and click a network interface.
2. In Related Tasks, click Add Channel.

To add a VLAN by using the Visualizer

1. Open the Network Visualizer, click the appliance to which you are logged on, and then do one of the following:
   - Click an existing VLAN, and then, in Related Tasks, click Add.
   - Click an existing bridge group, and then, in Related Tasks, click Add VLAN.

To add a bridge group by using the Visualizer

1. Open the Network Visualizer, click the appliance to which you are logged on, and then do one of the following:
   - Click an existing bridge group, and then, in Related Tasks, click Add.
   - Click an existing VLAN, and then, in Related Tasks, click Add Bridge Group.

To modify the settings of an interface or channel by using the Visualizer

1. Open the Network Visualizer and click the interface whose settings you want to modify.
2. In Related Tasks, click Open.

To enable or disable an interface or channel by using the Visualizer

1. Open the Network Visualizer and click the interface or channel that you want to enable or disable.
2. In Related Tasks, do one of the following.
   - To enable the interface or channel, click Enable.
   - To disable the interface or channel, click Disable.

To remove a configured channel, VLAN, or bridge group by using the Visualizer

1. Open the Network Visualizer and click the channel, VLAN, or bridge group that you want to remove from the configuration.
2. In Related Tasks, click Remove.

To view statistics for a node, channel, interface, or VLAN by using the Visualizer

1. Open the Network Visualizer and click the node, interface, or VLAN whose statistics you want to view.
2. In Related Tasks, click Statistics.

To set up an HA deployment by using the Visualizer

1. Open the Network Visualizer and click the appliance.
2. In Related Tasks, click HA Setup.

To force the secondary node to take over as the primary by using the Visualizer

1. Open the Network Visualizer and click one of the nodes.
2. In Related Tasks, click Force Failover.

To synchronize the secondary node's configuration with the primary node by using the Visualizer

1. Open the Network Visualizer and click one of the nodes.
2. In Related Tasks, click Force Synchronization.

To remove the peer node from the HA configuration

1. Open the Network Visualizer and click the peer node.
2. In Related Tasks, click Remove.

To copy the properties of a node or network entity by using the Visualizer

1. Open the Network Visualizer and click the appliance or network entity whose properties you want to copy to a document or spreadsheet.
2. In Related Tasks, click Copy Properties.

Access Control Lists

May 10, 2012

Access Control Lists (ACLs) filter IP traffic and secure your network from unauthorized access. An ACL is a set of conditions that the NetScaler ADC evaluates to determine whether to allow access. For example, the Finance department probably does not want to allow its resources to be accessed by other departments, such as HR and Documentation, and those departments want to restrict access to their data.

When the NetScaler ADC receives a data packet, it compares the information in the data packet with the conditions specified in the ACL and allows or denies access. The administrator of the organization can configure ACLs to function in the following processing modes:

- ALLOW—Process the packet.
- BRIDGE—Bridge the packet to the destination without processing it. The packet is directly sent by Layer 2 and Layer 3 forwarding.
- DENY—Drop the packet.

ACL rules are the first level of defense on the NetScaler ADC.

NetScaler supports the following types of ACLs:

- **Simple ACLs** filter packets on the basis of their source IP address and, optionally, their protocol, destination port, or traffic domain. Any packet that has the characteristics specified in the ACL is dropped.
- **Extended ACLs** filter data packets on the basis of various parameters, such as source IP address, source port, action, and protocol. An extended ACL defines the conditions that a packet must satisfy for the NetScaler ADC to process the packet, bridge the packet, or drop the packet.

Nomenclature

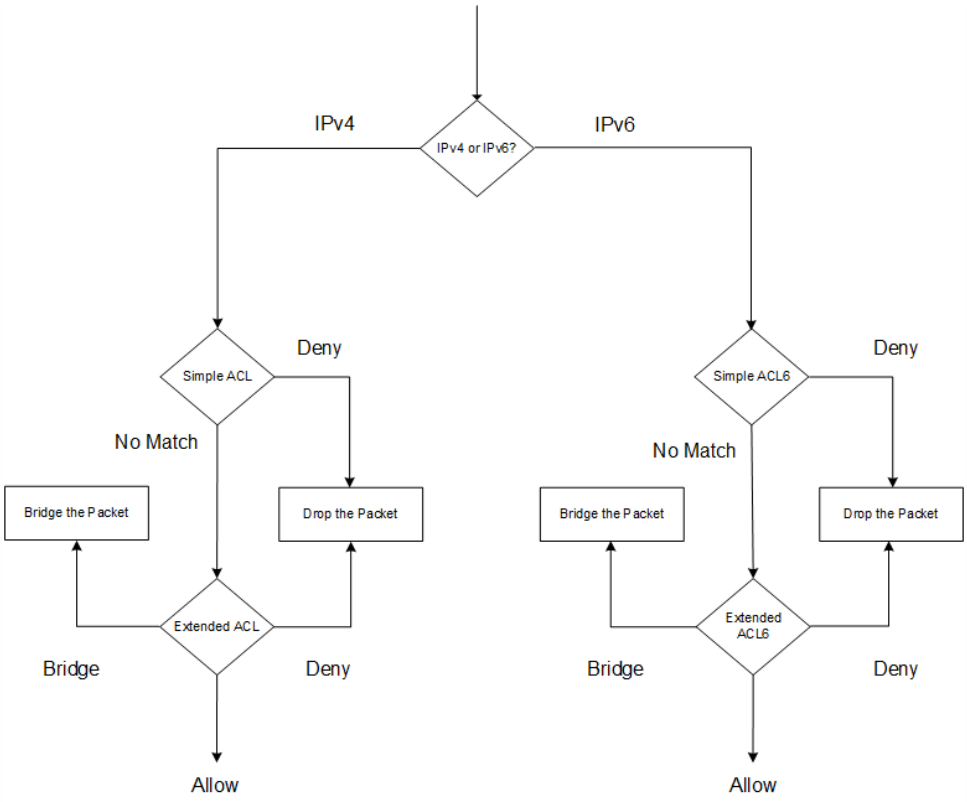
In the NetScaler user interfaces, the terms simple ACL and extended ACL refer to ACLs that process IPv4 packets. An ACL that processes IPv6 packets is called a simple ACL6 and or extended ACL6. When discussing both types, this documentation sometimes refers to both of them as simple ACLs or extended ACLs.

ACL Precedence

If both simple and extended ACLs are configured, incoming packets are compared to the simple ACLs first.

The NetScaler ADC first determines whether the incoming packet is an IPv4 or an IPv6 packet, and then compares the packet's characteristics to either simple ACLs or simple ACL6s. If a match is found, the packet is dropped. If no match is found, the packet is compared to extended ACLs or extended ACL6s. If that comparison results in a match, the packet is handled as specified in the ACL. The packet can be bridged, dropped, or allowed. If no match is found, the packet is allowed.

Figure 1. Simple and Extended ACLs Flow Sequence



Simple ACLs and Simple ACL6s

May 21, 2015

A simple ACL or simple ACL6 uses few parameters and can be configured only to drop IP packets. Packets can be dropped on the basis of their source IP address and, optionally, their protocol, destination port, or traffic domain.

When creating a simple ACL or simple ACL6, you can specify a time to live (TTL), in seconds, after which the ACL expires. ACLs with TTLs are not saved when you save the configuration. You can display simple ACLs and simple ACL6s to verify their configuration, and you can display their statistics.

This section includes the following details:

- Configuring Simple ACLs and Simple ACL6s
- Displaying Simple ACL and Simple ACL6 Statistics
- Terminating Established Connections

Configuring Simple ACLs and Simple ACL6s

Configuring a simple ACL or simple ACL6 on a NetScaler ADC can include the following tasks.

- Create simple ACLs or simple ACL6s to drop (deny) packets on the basis of their source IP address and, optionally, their protocol, destination port, or traffic domain.
- Remove simple ACLs or simple ACL6s. These ACLs cannot be modified once created. If you need to modify a simple ACL or simple ACL6, you must remove it and create a new one.

To create a simple ACL by using the command line interface

At the command prompt, type the following commands to add an ACL and verify the configuration:

- **add ns simpleacl** <aclname> DENY -**srcIP** <ip_addr> [-**destPort**<port> -**protocol** (TCP | UDP)] [-**TTL** <positive_integer>]
- **show ns simpleacl** [<aclname>]

Example

```
> add simpleacl rule1 DENY -srcIP 10.102.29.5 -TTL 600
Done
```

To create a simple ACL6 by using the command line interface

At the command prompt, type the following commands to add a simple ACL6 and verify the configuration:

- **add ns simpleacl6** <aclname> DENY -**srcIPv6** <ipv6_addr|null> [-**destPort**<port> -**protocol** (TCP | UDP)] [-**TTL** <positive_integer>]
- **show ns simpleacl6** [<aclname>]

Example

```
>  add ns simpleacl6 rule1 DENY –srcIPv6 3ffe:192:168:215::82 -destPort 80 -Protocol TCP -TTL 9000
 Done
```

To remove a single simple ACL by using the command line interface

At the command prompt, type:

- **rm ns simpleacl** <aclname>
- **show ns simpleacl**

To remove a single simple ACL6 by using the command line interface

At the command prompt, type:

- **rm ns simpleacl6**<aclname>
- **show ns simpleacl6**

To remove all simple ACLs by using the command line interface

At the command prompt, type:

- **clear ns simpleacl**
- **show ns simpleacl**

To remove all simple ACL6s by using the command line interface

At the command prompt, type:

- **clear ns simpleacl6**
- **show ns simpleacl6**

To create a simple ACL by using the configuration utility

Navigate to System > Network > ACLs and, on the **Simple ACLs** tab, add a new simple ACL.

To create a simple ACL6 by using the configuration utility

Navigate to System > Network > ACLs and, on the **Simple ACL6s** tab, add a new simple ACL6.

To remove a single simple ACL by using the configuration utility

Navigate to System > Network > ACLs and, on the **Simple ACLs** tab, delete the simple ACL.

To remove a single simple ACL6 by using the configuration utility

Navigate to System > Network > ACLs and, on the **Simple ACL6s** tab, delete the simple ACL6.

To remove all simple ACLs by using the configuration utility

1. Navigate to System > Network > ACLs.
2. On the **Simple ACLs** tab, in the **Action** list, click **Clear**.

To remove all simple ACL6s by using the configuration utility

1. Navigate to System > Network > ACLs.
2. On the **Simple ACL6s** tab, in the **Action** list, click **Clear**.

Displaying Simple ACL and Simple ACL6 Statistics

You can display the simple ACL (or simple ACL6) statistics, which include the number of hits, the number of misses, and the number of simple ACLs configured.

The following table describes statistics you can display for simple ACLs and simple ACL6s.

Statistic	Indicates
ACL hits	Packets matching an ACL
ACL misses	Packets not matching any ACL
ACL count	Number of ACLs configured

To display simple ACL statistics by using the command line interface

At the command prompt, type:

stat ns simpleacl
Example

```
> stat ns simpleacl

SimpleACL Statistics

                          Rate (/s)          Total
SimpleACL hits                    0              0
SimpleACL misses                  0          51872
SimpleACLs count                 --              2
Done
```

To display simple ACL6 statistics by using the command line interface

At the command prompt, type:

stat ns simpleacl6

To display simple ACL statistics by using the configuration utility

Navigate to System > Network > ACLs and, on the **Simple ACLs** tab, select the ACL and click **Statistics**.

To display simple ACL6 statistics by using the configuration utility

Navigate to System > Network > ACLs and, on the **Simple ACL6s** tab, select the simple ACL6 and click **Statistics**.

Terminating Established Connections

For a simple ACL or simple ACL6, the NetScaler ADC blocks any new connections that match the conditions specified in the ACL. Packets related to existing connections that were established before the ACL was created are not blocked. To terminate previously established connections that match an existing ACL, you can run a flush operation from the command

line interface or the configuration utility.

Flush can be useful in the following cases:

- You receive a list of blacklisted IP addresses and want to completely block those IP addresses from accessing the NetScaler ADC. In this case, you create simple ACLs or simple ACL6s to block any new connections from these IP addresses, and then flush any existing connections associated with those addresses.
- You want to terminate a large number of connections from a particular network without taking the time to terminate them one by one.

When you run flush, the NetScaler ADC searches through all of its established connections and terminates those that match conditions specified in any of the simple ACLs configured on the ADC.

Note: If you plan to create more than one simple ACL and flush existing connections that match any of them, you can minimize the effect on performance by first creating all of the simple ACLs and then running flush only once.

To terminate all established IPv4 connections that match any of your configured simple ACLs by using the command line interface

At the command prompt, type:

flush simpleacl -estSessions

To terminate all established IPv6 connections that match any of your configured simple ACL6s by using the command line interface

At the command prompt, type:

flush simpleacl6 -estSessions

To terminate all established IPv4 connections that match any of your configured simple ACLs by using the configuration utility

1. Navigate to System > Network > ACLs.
2. On the **Simple ACLs** tab, in the **Action** list, click **Flush**.

To terminate all established IPv6 connections that match any of your configured simple ACL6s by using the configuration utility

1. Navigate to System > Network > ACLs.
2. On the **Simple ACL6s** tab, in the **Action** list, click **Flush**.

Extended ACLs and ACL6s

Jan 31, 2011

Extended ACLs and extended ACL6s provide parameters and actions not available with simple ACLs. You can filter data on the basis of parameters such as source IP address, source port, action, and protocol. You can specify tasks to allow a packet, deny a packet, or bridge a packet.

Extended ACLs and ACL6s can be modified after they are created, and you can renumber their priorities to specify the order in which they are evaluated.

Note: If you configure both simple and extended ACLs, simple ACLs take precedence over extended ACLs.
The following actions can be performed on extended ACLs and ACL6s: Modify, Apply, Disable, Enable, Remove, and Renumber (the priority). You can display extended ACLs and ACL6s to verify their configuration, and you can display their statistics.

You can configure the NetScaler ADC to log details for packets that match an extended ACL. However, you cannot log details of packets that match an ext

Applying Extended ACLs and ACL6s

Unlike simple ACLs and ACL6s, extended ACLs and ACL6s created on the NetScaler ADC do not work until they are applied. Also, if you make any modifications to an extended ACL or ACL6, such as disabling the ACLs, changing a priority, or deleting the ACLs, you must reapply the extended ACLs or ACL6s. You must also reapply them after enabling logging. The procedure to apply extended ACLs or ACL6s reapplies all of them. For example, if you have applied extended ACL rules 1 through 10, and you then create and apply rule 11, the first 10 rules are applied afresh.

If a session has a DENY ACL related to it, that session is terminated when you apply the ACLs.

Extended ACLs and ACL6s are enabled by default. When they are applied, the NetScaler ADC starts comparing incoming packets against them. However, if you disable them, they are not used until you reenable them, even if they are reapplied.

Renumbering the priorities of Extended ACLs and ACL6s

Priority numbers determine the order in which extended ACLs or ACL6s are matched against a packet. An ACL with a lower priority number has a higher priority. It is evaluated before ACLs with higher priority numbers (lower priorities), and the first ACL to match the packet determines the action applied to the packet.

When you create an extended ACL or ACL6, the NetScaler ADC automatically assigns it a priority number that is a multiple of 10, unless you specify otherwise. For example, if two extended ACLs have priorities of 20 and 30, respectively, and you want a third ACL to have a value between those numbers, you might assign it a value of 25. If you later want to retain the order in which the ACLs are evaluated but restore their numbering to multiples of 10, you can use the renumber procedure.

This section includes the following details:

- Configuring Extended ACLs and ACL6s
- Logging Extended ACLs (IPv4 Only)
- Displaying Extended ACL and ACL6s Statistics
- Sample Configurations

Configuring Extended ACLs and ACL6s

Configuring an extended ACL or ACL6 on a NetScaler ADC consists of the following tasks.

- Create an extended ACL or ACL6 to either allow, deny, or bridge a packet. You can specify an IP address or range of IP addresses to match against the source or destination IP addresses of the packets. You can specify a protocol to match against the protocol of incoming packets.
- (Optional) You can modify extended ACLs or ACL6s that you previously created. Or, if you want to temporarily take one out of use you can disable it, and later reenable it.
- Apply extended ACLs or ACL6s. After you create, modify, disable or reenable, or delete an extended ACL or ACL6, you must apply the extended ACLs or ACL6s to activate them.
- (Optional) Renumber the priorities of extended ACLs or ACL6s. If you have configured ACLs with priorities that are not multiples of 10 and want to restore the numbering to multiples of 10, use the renumber procedure.

To create an extended ACL by using the command line interface

At the command prompt, type:

- **add ns acl** <aclname> <aclaction> [-**srcIP** [<operator>] <srcIPVal>] [-**srcPort** [<operator>] <srcPortVal>] [-**destIP** [<operator>] <destIPVal>] [-**destPort** [<operator>] <destPortVal>] [-**TTL** <positive_integer>] [-**srcMac** <mac_addr>] [(-**protocol** <protocol> [-established]) | -**protocolNumber** <positive_integer>] [-**vlan** <positive_integer>] [-**interface** <interface_name>] [-**icmpType** <positive_integer> [-**icmpCode** <positive_integer>]] [-**priority** <positive_integer>] [-**state** (ENABLED | DISABLED)] [-**logstate** (ENABLED | DISABLED) [-**ratelimit** <positive_integer>]]
- **show ns acl** [<aclName>]

Example

```
> add ns acl restrict DENY -srcport 45-1024 -destIP 192.168.1.1 -protocol TCP
 Done
```

To create an extended ACL6 by using the command line interface

At the command prompt, type:

- **add ns acl6** <acl6name> <acl6action> [-**srcIPv6** [<operator>] <srcIPv6Val>] [-**srcPort** [<operator>] <srcPortVal>] [-**destIPv6** [<operator>] <destIPv6Val>] [-**destPort** [<operator>] <destPortVal>] [-**TTL** <positive_integer>] [-**srcMac** <mac_addr>] [(-**protocol** <protocol> [-established]) | -**protocolNumber** <positive_integer>] [-**vlan** <positive_integer>] [-**interface** <interface_name>] [-**icmpType** <positive_integer> [-icmpCode <positive_integer>]] [-**priority** <positive_integer>] [-**state** (ENABLED | DISABLED)]
- **show ns acl6** [<aclName>]

Example

```
> add ns acl6 rule6  DENY -srcport 45-1024 -destIPv6 2001::45 -protocol TCP
 Done
```

To modify an extended ACL by using the command line interface

To modify an extended ACL, type the **set ns acl** command, the name of the extended ACL, and the parameters to be changed, with their new values.

To modify an extended ACL6 by using the command line interface

To modify an extended ACL6, type the **set ns acl** command, the name of the extended ACL6, and the parameters to be changed, with their new values.

To disable or enable an extended ACL by using the command line interface

At the command prompt, type one of the following commands:

- **disable ns acl** <aclname>
- **enable ns acl** <aclname>

To disable or enable an extended ACL6 by using the command line interface

At the command prompt, type one of the following commands:

- **disable ns acl6** <aclname>
- **enable ns acl6**<aclname>

To apply extended ACLs by using the command line interface

At the command prompt, type:

apply ns acls

To apply extended ACL6s by using the command line interface

At the command prompt, type:

apply ns acls6

To renumber the priorities of extended ACLs by using the command line interface

At the command prompt, type:

renumber ns acls

To renumber the priorities of extended ACL6s by using the command line interface

At the command prompt, type:

renumber ns acls6

To configure an extended ACL by using the configuration utility

Navigate to **System** > **Network** > **ACLs** and, on the **Extended ACLs** tab, add a new extended ACL or edit an existing extended ACL. To enable or disable an existing extended ACL, select it, and then select **Enable** or **Disable** from the **Action** list.

To configure an extended ACL6sACL6s by using the configuration utility

Navigate to **System** > **Network** > **ACLs** and, on the **ACL6s** tab, add a new extended ACL6 or edit an existing extended ACL6. To enable or disable an existing extended ACL6, select it, and then select **Enable** or **Disable** from the **Action** list.

To apply extended ACLs by using the configuration utility

Navigate to **System** > **Network** > **ACLs** and, on the **Extended ACLs** tab, in the **Action** list, click **Apply**.

To apply extended ACL6sACL6s by using the configuration utility

Navigate to **System** > **Network** > **ACLs** and, on the **ACL6s** tab, in the **Action** list, click **Apply**.

To renumber the priorities of extended ACLs by using the configuration utility

Navigate to **System** > **Network** > **ACLs** and, on the **Extended ACLs** tab, in the **Action** list, click **Renumber Priority (s)**.

To renumber the priorities of extended ACL6sACL6s by using the configuration utility

Navigate to **System** > **Network** > **ACLs** and, on the **ACL6s** tab, in the **Action** list, click **Renumber Priority (s)**.

Logging Extended ACLs (IPv4 Only)

You can configure the NetScaler ADC to log details for packets that match extended ACLs.

Note: You cannot enable logging for extended ACL6s.
In addition to the ACL name, the logged details include packet-specific information such as the source and destination IP addresses. The information is stored either in the syslog file or in the nslog file, depending on the type of global logging (syslog or nslog) enabled.

Logging must be enabled at both the global level and the ACL level. The global setting takes precedence. For more information about enabling logging globally, see "Audit Logging."

To optimize logging, when multiple packets from the same flow match an ACL, only the first packet's details are logged, and the counter is incremented for every packet that belongs to the same flow. A flow is defined as a set of packets that have the same values for the source IP address, destination IP address, source port, destination port, and protocol parameters. To avoid flooding of log messages, the NetScaler ADC performs internal rate limiting so that packets belonging to the same flow are not repeatedly logged. The total number of different flows that can be logged at any given time is limited to 10,000.

Note: You must apply ACLs after you enable logging.

To configure extended ACL Logging by using the command line interface

At the command prompt, type the following commands to configure logging and verify the configuration:

- **set ns acl** <aclName> [-**logState** (ENABLED | DISABLED)] [-**rateLimit** <positive_integer>]
- **show ns acl** [<aclName>]

Example

```
> set ns acl restrict -logstate ENABLED -ratelimit 120
Warning: ACL modified, apply ACLs to activate change
```

To configure extended ACL Logging by using the configuration utility

1. Navigate to System > Network > ACLs and, on the **Extended ACLs** tab, open the extended ACL.
2. Set the following parameters:
   - **Log State**—Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.
   - **Log Rate Limit**—Maximum number of log messages to be generated per second. If you set this parameter, you must enable the Log State parameter.

Displaying Extended ACL and ACL6s Statistics

You can display statistics of extended ACLs and ACL6s.

The following table lists the statistics associated with extended ACLs and ACL6s, and their descriptions.

Statistic	Specifies
Allow ACL hits	Packets matching ACLs with processing mode set to ALLOW. The NetScaler ADC processes these packets.
NAT ACL hits	Packets matching a NAT ACL, resulting in a NAT session.
Deny ACL hits	Packets dropped because they match ACLs with processing mode set to DENY.
Bridge ACL hits	Packets matching a bridge ACL, which in transparent mode bypasses service processing.
ACL hits	Packets matching an ACL.
ACL misses	Packets not matching any ACL.

To display the statistics of all extended ACLs by using the command line interface

At the command prompt, type:

stat ns acl

To display the statistics of all extended ACL6s by using the command line interface

At the command prompt, type:

stat ns acl6

To display the statistics of an extended ACL by using the configuration utility

Navigate to System > Network > ACLs, on the **Extended ACLs** tab, select the extended ACL, and click **Statistics**.

To display the statistics of an extended ACL6 by using the configuration utility

Navigate to System > Network > ACLs, on the **Extended ACL6s** tab, select the extended ACL, and click **Statistics**.

Sample Configurations

The following table shows examples of configuring extended ACL rules through the command line interface.

Action - ALLOW	
Tasks	**Steps**

Create an extended ACL rule to allow a particular host to access the servers.	>add ns acl allow-client ALLOW -srcIP = 40.40.40.1 Done
Create an extended ACL rule to allow a particular network to access the servers.	>add ns acl allow-client-net ALLOW -srcIP = 40.40.40.0-40.40.40.255 Done
Create extended ACL rules to allow HTTP, TFTP, and ICMP traffic.	>add acl allow-http ALLOW -protocol tcP -destport 80 Done Done >add acl allow-tftp ALLOW -protocol udp -destport 69 Done >add acl allow-icmp ALLOW -protocol icmp Done
Create an extended ACL rule to allow access to a particular destination/network.	>add acl allow-dest-access ALLOW -destip 20.20.20.0-20.20.20.255 Done
Create an extended ACL rule to allow traffic coming from a particular VLAN.	>add acl allow-vlan ALLOW -vlan 3000 Done
Action - DENY	
Tasks	**Steps**
Create an extended ACL rule to deny access to the servers by a particular host.	>add ns acl deny-client DENY -srcIP = 50.50.50.1 Done
Create an extended ACL rule to deny access to the servers from a particular network.	> add ns acl deny-client-net DENY -srcIP = 50.50.50.0-50.50.50.255 Done
Create extended ACL rules to deny Telnet and FTP traffic.	>add ns acl deny-client-Telnet DENY -protocol TCP -destPort 23

	Done > add ns acl deny-client-FTP DENY -protocol TCP -destPort 20-21 Done
Create an extended ACL rule to deny TCP traffic to port 80 from a particular host/network.	>add ns acl deny-client-TCP DENY -protocol TCP -destPort 80 -destIP 20.20.20.0-20.20.20.255 Done
Create an extended ACL rule to deny traffic from a particular VLAN.	> add acl deny-vlan DENY -vlan 2000 Done
Action - BRIDGE	
Tasks	**Steps**
Create an extended ACL rule to bridge FTP traffic.	>add ns acl bridge-ftp BRIDGE -protocol TCP -destport 21 Done >add ns acl bridge-ftp-data BRIDGE -protocol TCP -destport 21 Done
Create an extended ACL rule to bridge all traffic from a particular VLAN.	>add ns acl bridge-client-vlan BRIDGE -vlan 1000 Done
MAC Address Filtering	
Tasks	**Steps**
Create an extended ACL rule to allow traffic from a particular MAC address to a particular host.	>add ns acl allow-mac-host ALLOW -srcMAC 2a:c1:69:92:a0:7b -destIP 10.10.10.1 Done
ACL with RNAT (Typically, RNAT is used to allow servers configured with private non-routable IP addresses to initiate connections to the Internet.)	

Tasks	Steps
Create an RNAT rule for a particular host.	>add ns acl rnat-acl-host ALLOW -srcIP 40.40.40.1 Done >apply ns acls Done >set rnat rnat-acl Done
Create an RNAT rule for a particular network.	>add ns acl rnat-acl-network ALLOW -srcIP 40.40.40.0-40.40.40.255 Done >set rnat rnat-acl-network -NATIP 5.5.5.1 Done
ACL with Forwarding Session	
Create a forwarding session rule for a case in which a client request forwarded to a server results in a response that has to return by the same path.	>add ns acl forward-acl-host ALLOW -srcIP 20.20.20.1 Done >add forwardingSession fs -aclname forward-acl-host Done

IP Routing

Sep 06, 2013

NetScaler appliances support both dynamic and static routing. Because simple routing is not the primary role of a NetScaler, the main objective of running dynamic routing protocols is to enable route health injection (RHI), so that an upstream router can choose the best among multiple routes to a topographically distributed virtual server.

Most NetScaler implementations use some static routes to reduce routing overhead. You can create backup static routes and monitor routes to enable automatic switchover in the event that a static route goes down. You can also assign weights to facilitate load balancing among static routes, create null routes to prevent routing loops, and configure IPv6 static routes. You can configure policy based routes (PBRs), for which routing decisions are based on criteria that you specify.

This document includes the following information:

- Configuring Dynamic Routes
- Configuring Static Routes
- Configuring Policy-Based Routes
- Troubleshooting Routing Issues

Configuring Dynamic Routes

Sep 30, 2015

When a dynamic routing protocol is enabled, the corresponding routing process monitors route updates and advertises routes. Routing protocols enable an upstream router to use the equal cost multipath (ECMP) technique to load balance traffic to identical virtual servers hosted on two standalone NetScaler appliances. Dynamic routing on a NetScaler appliance uses three routing tables. In a high-availability setup, the routing tables on the secondary appliance mirror those on the primary.

For command reference guides and unsupported commands on dynamic routing protocol, see Dynamic Routing Protocol Command Reference Guides and Unsupported Commands.

The NetScaler supports the following protocols:

- Routing Information Protocol (RIP) version 2
- Open Shortest Path First (OSPF) version 2
- Border Gateway Protocol (BGP)
- Routing Information Protocol next generation (RIPng) for IPv6
- Open Shortest Path First (OSPF) version 3 for IPv6
- ISIS Protocol

You can enable more than one protocol simultaneously.

Routing Tables in the NetScaler

In a NetScaler appliance, the NetScaler kernel routing table, the FreeBSD kernel routing table, and the NSM FIB routing table each hold a different set of routes and serve a different purpose. They communicate with each other by using UNIX routing sockets. Route updates are not automatically propagated from one routing table to another. You must configure propagation of route updates for each routing table.

NS Kernel Routing Table

The NS kernel routing table holds subnet routes corresponding to the NSIP and to each SNIP and MIP. Usually, no routes corresponding to VIPs are present in the NS kernel routing table. The exception is a VIP added by using the **add ns ip** command and configured with a subnet mask other than 255.255.255.255. If there are multiple IP addresses belonging to the same subnet, they are abstracted as a single subnet route. In addition, this table holds a route to the loopback network (127.0.0.0) and any static routes added through the command line interface (CLI). The entries in this table are used by the NetScaler in packet forwarding. From the NetScaler CLI, they can be inspected with the **show route** command.

FreeBSD Routing Table

The sole purpose of the FreeBSD routing table is to facilitate initiation and termination of management traffic (telnet, ssh, etc.). In a NetScaler appliance, these applications are tightly coupled to FreeBSD, and it is imperative for FreeBSD to have the necessary information to handle traffic to and from these applications. This routing table contains a route to the NSIP subnet and a default route. In addition, FreeBSD adds routes of type WasCloned(W) when the NetScaler establishes connections to hosts on local networks. Because of the highly specialized utility of the entries in this routing table, all other route updates from the NS kernel and NSM FIB routing tables bypass the FreeBSD routing table. Do not modify it with the **route** command. The FreeBSD routing table can be inspected by using the **netstat** command from any UNIX shell.

Network Services Module (NSM) FIB

The NSM FIB routing table contains the advertisable routes that are distributed by the dynamic routing protocols to their peers in the network. It may contain:

Connected routes

IP subnets that are directly reachable from the NetScaler. Typically, routes corresponding to the NSIP subnet and subnets over which routing protocols are enabled are present in NSM FIB as connected routes.

Kernel routes

All the VIP addresses on which the -hostRoute option is enabled are present in NSM FIB as kernel routes if they satisfy the required RHI Levels. In addition, NSM FIB contains any static routes configured on the NetScaler CLI that have the -advertise option enabled. Alternatively, if the NetScaler is operating in Static Route Advertisement (SRADV) mode, all static routes configured on the NetScaler CLI are present in NSM FIB. These static routes are marked as kernel routes in NSM FIB, because they actually belong to the NS kernel routing table.

Static routes

Normally, any static route configured in VTYSH is present in NSM FIB. If administrative distances of protocols are modified, this may not always be the case. An important point to note is that these routes can never get into the NS kernel routing table.

Learned routes

If the NetScaler is configured to learn routes dynamically, the NSM FIB contains routes learned by the various dynamic routing protocols. Routes learned by OSPF, however, need special processing. They are downloaded to FIB only if the fib-install option is enabled for the OSPF process. This can be done from the router-config view in VTYSH.

High Availability Setup

In a high availability setup, the primary node runs the routing process and propagates routing table updates to the secondary node. The routing table of the secondary node mirrors the routing table on the primary node.

Non-Stop Forwarding

After failover, the secondary node takes some time to start the protocol, learn the routes, and update its routing table. But this does not affect routing, because the routing table on the secondary node is identical to the routing table on the primary node. This mode of operation is known as non-stop forwarding.

Black Hole Avoidance Mechanism

After failover, the new primary node injects all its VIP routes into the upstream router. However, that router retains the old primary node's routes for 180 seconds. Because the router is not aware of the failover, it attempts to load balance traffic between the two nodes. During the 180 seconds before the old routes expire, the router sends half the traffic to the old, inactive primary node, which is, in effect, a black hole.

To prevent this, the new primary node, when injecting a route, assigns it a metric that is slightly lower than the one specified by the old primary node.

Interfaces for Configuring Dynamic Routing

To configure dynamic routing, you can use either the configuration utility or a command-line interface. The NetScaler supports two independent command-line interfaces: the NetScaler CLI and the Virtual Teletype Shell (VTYSH). The

NetScaler CLI is the appliance's native shell. VTYSH is exposed by ZebOS. The NetScaler routing suite is based on ZebOS, the commercial version of GNU Zebra.

Note: Citrix recommends that you use VTYSH for all commands except those that can be configured only on the NetScaler CLI. Use of the NetScaler CLI should generally be limited to commands for enabling the routing protocols, configuring host route advertisement, and adding static routes for packet forwarding.

Dynamic Routing Protocol Command Reference Guides and Unsupported Commands

The following table lists command reference guide links, for various dynamic routing protocols, and unsupported commands on the NetScaler appliance:

Dynamic Routing Protocol	Command Reference Guide	Unsupported Commands
OSPF	OSPF Command Reference	• Domain-id command • Graceful restart related commands • OSPF-TE related commands • OSPF-VPN related commands • CSPF-TE related commands • ip ospf resync-timeout command • capability opaque command • enable ext-ospf-multi-inst command
IPv6 OSPF (OSPFv3)	OSPF Command Reference	• Graceful restart related commands • OSPF-TE related commands
BGP	BGP Command Reference	• VPN/VRF related commands • Graceful restart related commands • MPLS related commands • 6PE commands (IPv6 provider edge) • MD5 authentication related commands • Multicast options • set-overload-bit command
IS-IS	IS-IS Command Reference	• capability cspf command • enable-cspf command • mpls traffic-eng command • mpls traffic-eng router-id command • multi-topology for ipv6 address family related commands
RIP and IPv6 RIP (RIPng)	-	• neighbor command

Configuring RIP

Mar 20, 2012

Routing Information Protocol (RIP) is a Distance Vector protocol. The NetScaler supports RIP as defined in RFC 1058 and RFC 2453. RIP can run on any subnet.

After enabling RIP, you need to configure advertisement of RIP routes. For troubleshooting, you can limit RIP propagation. You can display RIP settings to verify the configuration.

Enabling and Disabling RIP

Updated: 2013-08-30

Use either of the following procedures to enable or disable RIP. After you enable RIP, the NetScaler appliance starts the RIP process. After you disable RIP, the appliance stops the RIP process.

To enable or disable RIP routing by using the command line interface

At the command prompt, enter one of the following commands to enable or disable RIP:

- enable ns feature RIP
- disable ns feature RIP

To enable or disable RIP routing by using the configuration utility

1. Navigate to System > Settings, in Modes and Features group, click Change advanced features.
2. Select or clear the RIP Routing option.

Advertising Routes

Updated: 2013-08-30

RIP enables an upstream router to load balance traffic between two identical virtual servers hosted on two standalone NetScaler appliances. Route advertisement enables an upstream router to track network entities located behind the NetScaler.

To configure RIP to advertise routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router rip	Start the RIP routing process and enter configuration mode for the routing process.

Command	Specifies
redistribute static	Redistribute static routes.
redistribute kernel	Redistribute kernel routes.

Example:

```
>VTYSH
NS# configure terminal
NS(config)# router rip
NS(config-router)# redistribute static
NS(config-router)# redistribute kernel
```

Limiting RIP Propagations

Updated: 2013-08-30

If you need to troubleshoot your configuration, you can configure listen-only mode on any given interface.

To limit RIP propagation by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router rip	Start the RIP routing process and enter configuration mode for the routing process.
passive-interface < vlan_name>	Suppress routing updates on interfaces bound to the specified VLAN.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router rip
NS(config-router)# passive-interface VLAN0
```

Verifying the RIP Configuration

Updated: 2013-08-30

You can display the routing table and other RIP settings.

To view the RIP settings by using the VTYSH command line

At the command prompt, type the following commands in the following order:

Command	Specifies
VTYSH	Display VTYSH command prompt.
sh rip	Display updated RIP routing table.
sh rip interface <vlan_name>	Displays RIP information for the specified VLAN.

Example

```
NS# VTYSH
NS# sh rip
NS# sh rip interface VLAN0
```

Configuring OSPF

Feb 07, 2014

The NetScaler supports Open Shortest Path First (OSPF) Version 2 (RFC 2328). The features of OSPF on the NetScaler are:

- If a vserver is active, the host routes to the vserver can be injected into the routing protocols.
- OSPF can run on any subnet.
- Route learning advertised by neighboring OSPF routers can be disabled on the NetScaler.
- The NetScaler can advertise Type-1 or Type-2 external metrics for all routes.
- The NetScaler can advertise user-specified metric settings for VIP routes. For example, you can configure a metric per VIP without special route maps.
- You can specify the OSPF area ID for the NetScaler.
- The NetScaler supports not-so-stubby-areas (NSSAs). An NSSA is similar to an OSPF stub area but allows injection of external routes in a limited fashion into the stub area. To support NSSAs, a new option bit (the N bit) and a new type (Type 7) of Link State Advertisement (LSA) area have been defined. Type 7 LSAs support external route information within an NSSA. An NSSA area border router (ABR) translates a type 7 LSA into a type 5 LSA that is propagated into the OSPF domain. The OSPF specification defines only the following general classes of area configuration:
  - Type 5 LSA: Originated by routers internal to the area are flooded into the domain by AS boarder routers (ASBRs).
  - Stub: Allows no type 5 LSAs to be propagated into/throughout the area and instead depends on default routing to external destinations.

After enabling OSPF, you need to configure advertisement of OSPF routes. For troubleshooting, you can limit OSPF propagation. You can display OSPF settings to verify the configuration.

Enabling and Disabling OSPF

Updated: 2013-09-05

To enable or disable OSPF, you must use either the command line interface or the configuration utility. When OSPF is enabled, the NetScaler starts the OSPF process. When OSPF is disabled, the NetScaler stops the OSPF routing process.

To enable or disable OSPF routing by using the command line interface

At the command prompt, type one of the following commands:

1. enable ns feature OSPF
2. disable ns feature OSPF

To enable or disable OSPF routing by using the configuration utility

1. Navigate to System > Settings, in Modes and Features group, click Change advanced features.
2. Select or clear the OSPF Routing option.

Advertising OSPF Routes

Updated: 2013-08-30

OSPF enables an upstream router to load balance traffic between two identical virtual servers hosted on two standalone NetScaler appliances. Route advertising enables an upstream router to track network entities located behind the NetScaler.

To configure OSPF to advertise routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enters global configuration mode.
router OSPF	Start OSPF routing process and enter configuration mode for the routing process.
network A.B.C.D/M area <0-4294967295>	Enable routing on an IP network.
redistribute static	Redistribute static routes.
redistribute kernel	Redistribute kernel routes.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router OSPF
NS(config-router)# network 10.102.29.0/24 area 0
NS(config-router)# redistribute static
NS(config-router)# redistribute kernel
```

Limiting OSPF Propagations

Updated: 2013-08-30

If you need to troubleshoot your configuration, you can configure listen-only mode on any given VLAN.

To limit OSPF propagation by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router OSPF	Start OSPF routing process and enters configuration mode for the routing process.

Command	Specifies
passive-interface < vlan_name>	Suppress routing updates on interfaces bound to the specified VLAN.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router OSPF
NS(config-router)# passive-interface VLAN0
```

Verifying the OSPF Configuration

Updated: 2013-08-30

You can display current OSPF neighbors, and OSPF routes.

To view the OSPF settings by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
sh OSPF neighbor	Displays current neighbors.
sh OSPF route	Displays OSPF routes.

Example

```
>VTYSH
NS# sh OSPF neighbor
NS# sh OSPF route
```

Configuring BGP

Mar 20, 2012

The NetScaler appliance supports BGP (RFC 4271). The features of BGP on the NetScaler are:

- The NetScaler advertises routes to BGP peers.
- The NetScaler injects host routes to virtual IP addresses (VIPs), as determined by the health of the underlying virtual servers.
- The NetScaler generates configuration files for running BGP on the secondary node after failover in an HA configuration.
- This protocol supports IPv6 route exchanges.

After enabling BGP, you need to configure advertisement of BGP routes. For troubleshooting, you can limit BGP propagation. You can display BGP settings to verify the configuration.

Prerequisites for IPv6 BGP

Before you begin configuring IPv6 BGP, do the following:

- Make sure that you understand the IPv6 BGP protocol.
- Install the IPv6PT license on the NetScaler appliance.
- After installing the IPv6PT license, enable the IPv6 feature.

Enabling and Disabling BGP

Updated: 2013-09-05

To enable or disable BGP, you must use either the command line interface or the configuration utility. When BGP is enabled, the NetScaler appliance starts the BGP process. When BGP is disabled, the appliance stops the BGP process.

To enable or disable BGP routing by using the command line interface

At the command prompt, type one of the following commands:

- enable ns feature BGP
- disable ns feature BGP

To enable or disable BGP routing by using the configuration utility

1. Navigate to System > Settings, in Modes and Features group, click Change advanced features.
2. Select or clear the BGP Routing option.

Advertising IPv4 Routes

Updated: 2013-08-30

You can configure the NetScaler appliance to advertise host routes to VIPs and to advertise routes to downstream networks.

To configure BGP to advertise IPv4 routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router BGP < ASnumber>	BGP autonomous system. < ASnumber> is a required parameter. Possible values: 1 to 4,294,967,295.
Neighbor < IPv4 address> remote-as < as-number>	Update the IPv4 BGP neighbor table with the link local IPv4 address of the neighbor in the specified autonomous system.
Address-family ipv4	Enter address family configuration mode.
Neighbor < IPv4 address> activate	Exchange prefixes for the IPv4 router family between the peer and the local node by using the link local address.
redistribute kernel	Redistribute kernel routes.
redistribute static	Redistribute static routes.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router BGP 5
NS(config-router)# Neighbor a1bc::102 remote-as 100
NS(config-router)# Address-family ipv4
NS(config-router-af)# Neighbor 10.102.29.170 activate
NS(config-router)# redistribute kernel
NS(config-router)# redistribute static
```

Advertising IPv6 BGP Routes

Updated: 2013-08-30

Border Gateway Protocol (BGP) enables an upstream router to load balance traffic between two identical virtual servers hosted on two standalone NetScaler appliances. Route advertising enables an upstream router to track network entities located behind the NetScaler.

To configure BGP to advertise IPv6 routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router BGP < ASnumber>	BGP autonomous system. < ASnumber> is a required parameter. Possible values: 1 to 4,294,967,295.
Neighbor < IPv6 address> remote-as < as-number>	Update the IPv6 BGP neighbor table with the link local IPv6 address of the neighbor in the specified autonomous system.
Address-family ipv6	Enter address family configuration mode.
Neighbor < IPv6 address> activate	Exchange prefixes for the IPv6 router family between the peer and the local node by using the link local address.
redistribute kernel	Redistribute kernel routes.
redistribute static	Redistribute static routes.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router BGP 5
NS(config-router)# Neighbor a1bc::102 remote-as 100
NS(config-router)# Address-family ipv6
NS(config-router-af)# Neighbor a1bc::102 activate
NS(config-router)# redistribute kernel
NS(config-router)# redistribute static
```

Verifying the BGP Configuration

Updated: 2013-08-30

You can use VTYSH to display BGP settings.

To view the BGP settings using the VTYSH command line

At the command prompt, type:

```
VTYSH
You are now in the VTYSH command prompt. An output similar to the following appears:
NS170#
At the VTYSH command prompt, type:
```

```
NS170# sh ip BGP
NS170# sh BGP
NS170# sh ip BGP neighbors
NS170# sh ip BGP summary
NS170# sh ip BGP route-map <map-tag>
```

Configuring IPv6 RIP

Mar 20, 2012

IPv6 Routing Information Protocol (RIP) or RIPng is a Distance Vector protocol. This protocol is an extension of RIP to support IPv6. After enabling IPv6 RIP, you need to configure advertisement of IPv6 RIP routes. For troubleshooting, you can limit IPv6 RIP propagation. You can display IPv6 RIP settings to verify the configuration.

Prerequisites for IPv6 RIP

Before you begin configuring IPv6 RIP, do the following:

- Make sure that you understand the IPv6 RIP protocol.
- Install the IPv6PT license on the NetScaler appliance.
- Enable the IPv6 feature.

Enabling IPv6 RIP

Updated: 2013-08-30

You can enable or disable IPv6 RIP by using VTYSH. After you enable IPv6 RIP, the NetScaler starts the IPv6 RIP daemon. After you disable IPv6 RIP, the NetScaler stops the RIP daemon.

To enable IPv6 RIP by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
ns IPv6-routing	Start IPv6 dynamic routing daemon.
interface < vlan_name>	Enter VLAN configuration mode.
router ipv6 RIP	Start IPv6 RIP routing process on the VLAN.

Example

```
> VTYSH
NS# configure terminal
NS(config)# ns IPv6-routing
NS(config)# interface vlan0
NS(config-if)# router ipv6 RIP
```

Advertising IPv6 RIP Routes

Updated: 2013-08-30

IPv6 RIP enables an upstream router to load balance traffic between two identical vservers hosted on two standalone NetScaler devices. Route advertisement enables an upstream router to track network entities located behind the NetScaler.

To configure IPv6 RIP to advertise IPv6 routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router ipv6 rip	Start IPv6 RIP routing process and enter configuration mode for the routing process.
redistribute static	Redistribute static routes.
redistribute kernel	Redistribute kernel routes.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router ipv6 rip
NS(config-router)# redistribute static
NS(config-router)# redistribute kernel
```

Limiting IPv6 RIP Propagations

Updated: 2013-08-30

If you need to troubleshoot your configuration, you can configure the listen-only mode on any given interface.

To limit IPv6 RIP propagation by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.

Command	Specifies
router ipv6 rip	Start IPv6 RIP routing process and enter configuration mode for the routing process.
passive-interface < vlan_name>	Suppress routing updates on interfaces bound to the specified VLAN.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router ipv6 rip
NS(config-router)# passive-interface VLAN0
```

Verifying the IPv6 RIP Configuration

Updated: 2013-08-30

You can use VTYSH to display the IPv6 RIP routing table and IPv6 RIP information for a specified VLAN.

To view the IPv6 RIP settings by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Commands	Specifies
VTYSH	Display VTYSH command prompt.
sh ipv6 rip	Display updated IPv6 RIP routing table.
sh ipv6 rip interface <vlan_name>	Display IPv6 RIP information for the specified VLAN.

Example

```
NS# VTYSH
NS# sh ipv6 rip
NS# sh ipv6 rip interface VLAN0
```

Configuring IPv6 OSPF

Mar 20, 2012

IPv6 OSPF or OSPF version 3 (OSPF v3) is a link state protocol that is used to exchange IPv6 routing information. After enabling IPv6 OSPF, you need to configure advertisement of IPv6 OSPF routes. For troubleshooting, you can limit IPv6 OSPF propagation. You can display IPv6 OSPF settings to verify the configuration.

Prerequisites for IPv6 OSPF

Before you begin configuring IPv6 OSPF, do the following:

- Make sure that you understand the IPv6 OSPF protocol.
- Install the IPv6PT license on the NetScaler appliance.
- Enable the IPv6 feature.

Enabling IPv6 OSPF

Updated: 2013-08-30

To enable IPv6 OSPF, you must use the VTYSH command line. When IPv6 OSPF is enabled, the NetScaler appliance starts the IPv6 OSPF daemon. When IPv6 OSPF is disabled, the appliance stops the IPv6 OSPF daemon.

To enable IPv6 OSPF by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
ns IPv6-routing	Start IPv6 dynamic routing process.
interface < vlan_name>	Enter the VLAN configuration mode.
ipv6 router OSPF area <area-id>	Start IPv6 OSPF routing process on a VLAN.

Example

```
>VTYSH
NS# configure terminal
NS(config)# ns IPv6-routing
NS(config)# interface vlan0
NS(config-if)# ipv6 router OSPF area 3
```

Advertising IPv6 Routes

Updated: 2013-08-30

IPv6 OSPF enables an upstream router to load balance traffic between two identical vservers hosted on two standalone NetScaler devices. Route advertising enables an upstream router to track network entities located behind the NetScaler.

To configure IPv6 OSPF to advertise IPv6 routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Commands	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
router ipv6 OSPF	Start IPv6 OSPF routing process and enter configuration mode for the routing process.
redistribute static	Redistribute static routes.
redistribute kernel	Redistribute kernel routes.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router ipv6 OSPF
NS(config-router)# redistribute static
NS(config-router)# redistribute kernel
```

Limiting IPv6 OSPF Propagations

Updated: 2013-08-30

If you need to troubleshoot your configuration, you use VTYSH to configure listen-only mode on any given VLAN.

To limit IPv6 OSPF propagation by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Commands	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.

Commands	Specifies
router ipv6 OSPF	Start IPv6 OSPF routing process and enter configuration mode for the routing process.
passive-interface < vlan_name >	Suppress routing updates on interfaces bound to the specified VLAN.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router ipv6 OSPF
NS(config-router)# passive-interface VLAN0
```

Verifying the IPv6 OSPF Configuration

Updated: 2013-08-30

You use VTYSH to display IPv6 OSPF current neighbors and IPv6 OSPF routes.

To view the IPv6 OSPF settings by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Specifies
VTYSH	Display VTYSH command prompt.
sh ipv6 OSPF neighbor	Display current neighbors.
sh ipv6 OSPF route	Display IPv6 OSPF routes.

Example

```
>VTYSH
NS# sh ipv6 OSPF neighbor
NS# sh ipv6 OSPF route
```

Configuring ISIS

Aug 28, 2013

The NetScaler appliance supports the Intermediate System-to-Intermediate System (IS-IS or ISIS) dynamic routing protocol. This protocol supports IPv4 as well as IPv6 route exchanges. IS-IS is a link state protocol and is therefore less prone to routing loops. With the advantages of faster convergence and the ability to support larger networks, ISIS can be very useful in Internet Service Provider (ISP) networks.

Prerequisites for configuring ISIS

Before you begin configuring ISIS, do the following:

- Make sure that you understand the ISIS protocol.
- For IPV6 routes, enable:
  - IPv6 protocol translation feature.
  - IPv6 Dynamic Routing option on the VLANs on which you want to run ISIS protocol.

Enabling ISIS

Updated: 2013-08-30

Use either of the following procedures to enable the ISIS routing feature on the NetScaler appliance.

To enable ISIS routing by using the command line interface

At the command prompt, type:

```
enable ns feature ISIS
```

To enable ISIS routing by using the configuration utility

1. Navigate to System > Settings, in Modes and Features group, click Change advanced features.
2. Select or clear the ISIS Routing option.

Creating an ISIS Routing Process and Starting It on a VLAN

Updated: 2013-08-30

To create an ISIS routing process, you must use the VTYSH command line.

At the command prompt, type the following commands, in the order shown:

Command	Description
VTYSH	Displays VTYSH command prompt.
configure terminal	Enters the global configuration mode.
router ISIS [tag]	Creates an ISIS routing process and configuration mode for the routing process.

Command	Description
net XX....XXXX.YYYY.YYYY.YYYY.00	Specifies a NET value for the routing process, where: ● · XX. .. .XXXX is the Area Address (can be 1-13 bytes) ● · YYYY.YYYY.YYYY is the System ID (6 bytes) ● · 00 is the N-selector (1 byte) A NET value can be 8 to 20 bytes in length. The last byte is always the n-selector, and must be zero. The n-selector indicates that there is no transport entity and means that the packet is for the routing software of the appliance. The six bytes directly preceding the n-selector are the system ID. The system ID length is fixed and cannot be changed. The system ID must be unique throughout each area (Level 1) and throughout the backbone (Level 2). The bytes preceding the system ID are the area ID, which can be from 1 to 13 bytes in length. A maximum of three NETs per routing process are allowed with different area ID, but the system ID should be the same for all NETs.
is-type (level-1|level-1-2|level-2-only)	Sets the ISIS routing process to the specified level of routing. Default: level-1-2.
ns IPv6-routing	Starts the IPv6 dynamic routing daemon.
interface <vlan_name>	Enters the VLAN configuration mode.
ip router ISIS	Enables the ISIS routing process on the VLAN for IPv4 route exchanges.
ipv6 router ISIS	Enables the ISIS routing process on the VLAN for IPv6 route exchanges.

Example

```
> VTYSH
NS# configure terminal
NS(config)# router isis 11
NS(config-router)# net 15.aabb.ccdd.0097.00
NS(config-router)# is-type level-1
NS(config-router)# exit
NS(config)# ns IPv6-routing
NS(config)# interface vlan0
NS(config-if)#  ip router isis 11
NS(config-if)# ipv6 router isis 11
```

Advertising Routes

Updated: 2013-08-30

Route advertisement enables an upstream router to track network entities located behind the NetScaler appliance.

To configure ISIS to advertise routes by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Description

Command	Description
VTYSH	Displays the VTYSH command prompt.
configure terminal	Enters the global configuration mode.
router ISIS [tag]	Starts the ISIS routing instance and enter configuration mode for the routing process.
redistribute connected (level-1|level-1-2|level-2)	Redistributes connected routes, where • **level-1** : Redistribute connected routes into Level-1. • **level-1-2** : Redistribute connected routes into Level-1 and Level-2. • **level-2** : Redistribute connected routes into Level-2.
redistribute kernel(level-1|level-1-2|level-2)	Redistributes kernel routes, where: • **level-1** : Redistribute kernel routes into Level-1. • **level-1-2** : Redistribute kernel routes into Level-1 and Level-2. • **level-2** : Redistribute kernel routes into Level-2.

Example

```
>VTYSH
NS# configure terminal
NS(config)# router isis 11
NS(config-router)# redistribute connected level-1
NS(config-router)# redistribute kernel level-1
```

Limiting ISIS Propagations

Updated: 2013-08-30

If you need to troubleshoot your configuration, you can configure the listen-only mode on any given VLAN.

To limit ISIS propagation by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Command	Description
VTYSH	Displays the VTYSH command prompt.
configure terminal	Enters the global configuration mode.
router isis [tag]	Enters the configuration mode for the routing process.
passive-interface <vlan_name>	Suppresses routing updates on interfaces bound to the specified VLAN.

Example

>VTYSH
NS# configure terminal
NS(config)# router isis 11
NS(config-router)# passive-interface VLAN0
Verifying the ISIS Configuration

Updated: 2013-08-30

You can use VTYSH to display the ISIS routing table and ISIS information for a specified VLAN.

To view the ISIS settings by using the VTYSH command line

At the command prompt, type the following commands, in the order shown:

Commands	Description
VTYSH	Displays the VTYSH command prompt.
show ip isis route	Displays updated IPv4 ISIS routing table.
show ipv6 isis route	Displays updated IPv6 ISIS routing table.
sh isis interface <vlan_name>	Displays IPv6 ISIS information for the specified VLAN.

Example
NS# VTYSH
NS# show ip isis route
NS# show ipv6 isis route
NS# sh isis interface VLAN0

Installing Routes to the NetScaler Routing Table

Aug 30, 2013

The NetScaler appliance can use routes learned by various routing protocols after you install the routes in the appliance's routing table.

To install various routes to the internal routing table by using the VTYSH command line

At the command prompt, type the following commands as appropriate for the routes that you want to install:

Commands	Specifies
VTYSH	Display VTYSH command prompt.
configure terminal	Enter global configuration mode.
ns route-install Default	Install IPv4 default routes to the internal routing table.
ns route-install RIP	Install IPv4 RIP specific routes to the internal routing table.
ns route-install BGP	Install IPv4 BGP specific routes to the internal routing table.
ns route-install OSPF	Install IPv4 OSPF specific routes to the internal routing table.
ns route-install IPv6 Default	Install IPv6 default routes to the internal routing table.
ns route-install IPv6 RIP	Install IPv6 RIP specific routes to the internal routing table.
ns route-install IPv6 BGP	Install IPv6 BGP specific routes to the internal routing table.
ns route-install IPv6 OSPF	Install IPv6 OSPF specific routes to the internal routing table.

Example

```
>VTYSH
NS# configure terminal
NS# ns route-install Default
NS(config)# ns route-install RIP
NS(config)# ns route-install BGP
NS(config)# ns route-install OSPF
NS# ns route-install IPv6 Default
NS(config)# ns route-install IPv6 RIP
```

```
NS(config)# ns route-install IPv6 BGP
NS(config)# ns route-install IPv6 OSPF
```

Configuring Static Routes

Sep 06, 2013

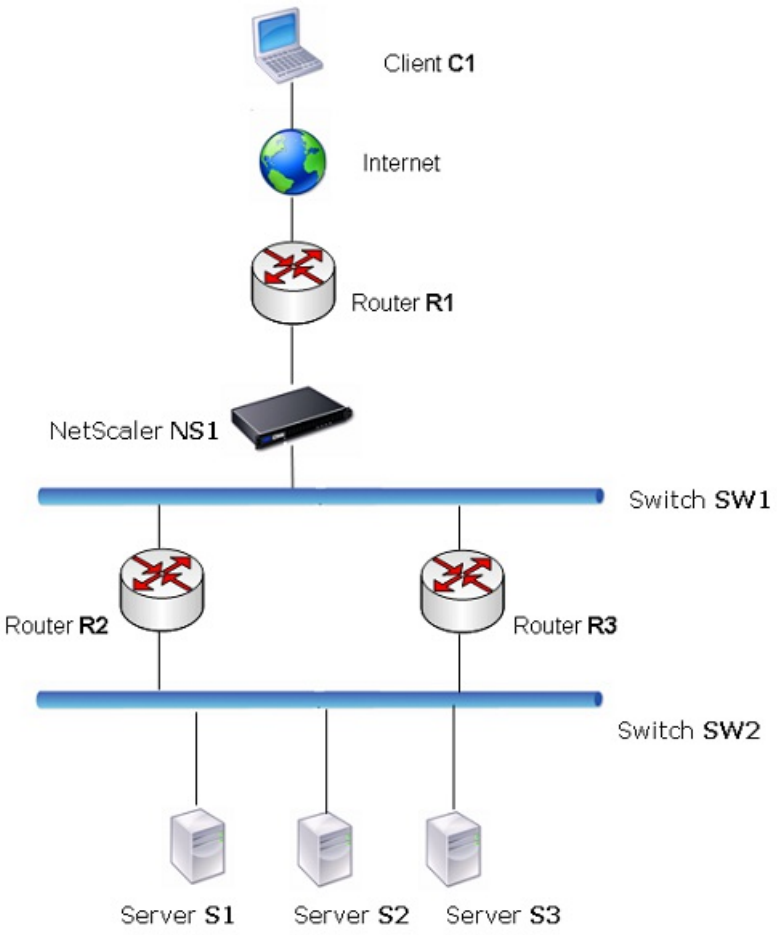
Static routes are manually created to improve the performance of your network. You can monitor static routes to avoid service disruptions. Also, you can assign weights to ECMP routes, and you can create null routes to prevent routing loops.

Monitored Static Routes

If a manually created (static) route goes down, a backup route is not automatically activated. You must manually delete the inactive primary static route. However, if you configure the static route as a monitored route, the NetScaler appliance can automatically activate a backup route.

Static route monitoring can also be based on the accessibility of the subnet. A subnet is usually connected to a single interface, but it can be logically accessed through other interfaces. Subnets bound to a VLAN are accessible only if the VLAN is up. VLANs are logical interfaces through which packets are transmitted and received by the NetScaler. A static route is marked as DOWN if the next hop resides on a subnet that is unreachable.

Note: In a high availability (HA) setup, the default value for monitored state routes (MSRs) on the secondary node is UP. The value is set to avoid a state transition gap upon failover, which could result in dropping packets on those routes.
Consider the following simple topology, in which a NetScaler is load balancing traffic to a site across multiple servers.



Router R1 moves traffic between the client and the NetScaler appliance. The appliance can reach servers S1 and S2 through routers R2 or R3. It has two static routes through which to reach the servers' subnet, one with R2 as the gateway and another with R3 as the gateway. Both these routes have monitoring enabled. The administrative distance of the static route with gateway R2 is lower than that of the static route with gateway R3. Therefore, R2 is preferred over R3 to forward traffic to the servers. Also, the default route on the NetScaler points to R1 so that all Internet traffic exits properly.

If R2 fails while monitoring is enabled on the static route, which uses R2 as the gateway, the NetScaler marks it as DOWN. The NetScaler now uses the static route with R3 as the gateway and forwards the traffic to the servers through R3.

The NetScaler supports monitoring of IPv4 and IPv6 static routes. You can configure the NetScaler to monitor an IPv4 static route either by creating a new ARP or PING monitor or by using existing ARP or PING monitors. You can configure the NetScaler to monitor an IPv6 static route either by creating a new Neighbor discovery for IPv6 (ND6) or PING monitor or by using the existing ND6 or PING monitors.

Weighted Static Routes

When the NetScaler appliance makes routing decisions involving routes with equal distance and cost, that is, Equal Cost Multi-Path (ECMP) routes, it balances the load between them by using a hashing mechanism based on the source and destination IP addresses. For an ECMP route, however, you can configure a weight value. The NetScaler then uses both the weight and the hashed value for balancing the load.

Null Routes

If the route chosen in a routing decision is inactive, the NetScaler appliance chooses a backup route. If all the backup routes become inaccessible, the appliance might reroute the packet to the sender, which could result in a routing loop leading to network congestion. To prevent this situation, you can create a null route, which adds a null interface as a gateway. The null route is never the preferred route, because it has a higher administrative distance than the other static routes. But it is selected if the other static routes become inaccessible. In that case, the appliance drops the packet and prevents a routing loop.

This section includes the following details:

- Configuring IPv4 Static Routes
- Configuring IPv6 Static Routes

Configuring IPv4 Static Routes

Updated: 2013-09-06

You can add a simple static route or a null route by setting a few parameters, or you can set additional parameters to configure a monitored or monitored and weighted static route. You can change the parameters of a static route. For example, you might want to assign a weight to an unweighted route, or you might want to disable monitoring on a monitored route.

To create a static route by using the command line interface

At the command prompt, type the following commands to create a static route and verify the configuration:

- add route <network> <netmask> <gateway>[-cost <positive_integer>] [-advertise (DISABLED | ENABLED)]
- show route [<network> <netmask> [<gateway>]] [<routeType>] [-detail]

Example

```
> add route 10.102.29.0 255.255.255.0 10.102.29.2 -cost 2 -advertise ENABLED
 Done
```

To create a monitored static route by using the command line interface

At the command prompt, type the following commands to create a monitored static route and verify the configuration:

- add route <network> <netmask> <gateway> [-distance <positive_integer>] [-weight <positive_integer>][-msr (ENABLED | DISABLED) [-monitor <string>]]
- show route [<network> <netmask> [<gateway>]] [<routeType>] [-detail]

Example

```
> add route 10.102.29.0 255.255.255.0 10.102.29.3 -distance 5 -weight 6 -msr ENBLED -monitor PING
 Done
```

To create a null route by using the command line interface

At the command prompt type:

- add route <network> <netmask> null
- show route <network> <netmask>

Example

```
> add route 10.102.29.0 255.255.255.0 null
 Done
```

To remove a static route by using the command line interface

At the command prompt, type:

rm route <network> <netmask> <gateway>

Example

```
> rm route 10.102.29.0 255.255.255.0 10.102.29.3
 Done
```

To configure a static route by using the configuration utility

Navigate to System > Network > Routes and, on the Basic tab, add a new static route, or edit an existing static route.

To remove a route by using the configuration utility

Navigate to System > Network > Routes and, on the Basic tab, delete the static route.

Configuring IPv6 Static Routes

Updated: 2013-09-06

You can configure a maximum of six default IPv6 static routes. IPv6 routes are selected on the basis of whether the MAC address of the destination device is reachable. This can be determined by using the IPv6 Neighbor Discovery feature. Routes are load balanced and only source/destination-based hash mechanisms are used. Therefore, route selection mechanisms such as round robin are not supported. The next hop address in the default route need not belong to the NSIP subnet.

To create an IPv6 route by using the command line interface

At the command prompt, type the following commands to create an IPv6 route and verify the configuration:

- add route6 <network> <gateway> [-vlan <positive_integer>]
- show route6 [<network> [<gateway>]

Example

```
> add route6 ::/0 FE80::67 -vlan 5
 Done
```

To create a monitored IPv6 static route by using the command line interface

At the command prompt, type the following commands to create a monitored IPv6 static route and verify the

configuration:

- add route6 <network> <gateway> [-msr (ENABLED | DISABLED) [-monitor <string>]
- show route6 [<network> [<gateway>]

Example

```
> add route6 ::/0 2004::1 -msr ENABLED -monitor PING
 Done
```

To remove an IPv6 route by using the command line interface

At the command prompt, type:

rm route6 <network> <gateway>

Example

```
> rm route6 ::/0 FE80::67
 Done
```

To configure an IPv6 route by using the configuration utility

Navigate to System > Network > Routes and, on the IPV6 tab, add a new IPv6 route, or edit an existing IPv6 route.

To remove an IPv6 route by using the configuration utility

Navigate to System > Network > Routes and, on the IPV6 tab, delete the IPv6 route.

Configuring Policy-Based Routes

Jun 11, 2012

Policy-based routing bases routing decisions on criteria that you specify. A policy-based route (PBR) specifies criteria for selecting packets and, typically, a next hop to which to send the selected packets. For example, you can configure the NetScaler appliance to route outgoing packets from a specific IP address or range to a particular next hop router. Each packet is matched against each configured PBR, in the order determined by the specified priorities, until a match is found. If no match is found, or if the matching PBR specifies a DENY action, the NetScaler applies the routing table for normal destination-based routing.

A PBR bases routing decisions for the data packets on parameters such as source IP address, source port, destination IP address, destination port, protocol, and source MAC address. A PBR defines the conditions that a packet must satisfy for the NetScaler to route the packet. These actions are known as "processing modes." The processing modes are:

- ALLOW - The NetScaler sends the packet to the designated next-hop router.
- DENY - The NetScaler applies the routing table for normal destination-based routing.

You can create PBRs for outgoing IPv4 and IPv6 traffic.

Many users begin by creating PBRs and then modifying them. To activate a new PBR, you must apply it. To deactivate a PBR, you can either remove or disable it. You can change the priority number of a PBR to give it a higher or lower precedence.

This document includes the following information:

- Configuring a Policy-Based Routes (PBR) for IPv4 Traffic
- Configuring a Policy-Based Routes (PBR6) for IPv6 Traffic

Configuring a Policy-Based Routes (PBR) for IPv4 Traffic

Mar 20, 2012

Configuring PBRs involves the following tasks:

- Create a PBR.
- Apply PBRs.
- (Optional) Disable or enable a PBR.
- (Optional) Renumber the priority of the PBR.

Creating or Modifying a PBR

Updated: 2013-10-31

You cannot create two PBRs with the same parameters. If you attempt to create a duplicate, an error message appears.

You can configure the priority of a PBR. The priority (an integer value) defines the order in which the NetScaler appliance evaluates PBRs. When you create a PBR without specifying a priority, the NetScaler automatically assigns a priority that is a multiple of 10.

If a packet matches the condition defined by the PBR, the NetScaler performs an action. If the packet does not match the condition defined by the PBR, the NetScaler compares the packet against the PBR with the next highest priority.

Instead of sending the selected packets to a next hop router, you can configure the PBR to send them to a link load balancing virtual server to which you have bound multiple next hops. This configuration can provide a backup if a next hop link fails.

Consider the following example. Two PBRs, p1 and p2, are configured on the NetScaler and automatically assigned priorities 20 and 30. You need to add a third PBR, p3, to be evaluated immediately after the first PBR, p1. The new PBR, p3, must have a priority between 20 and 30. In this case, you can specify the priority as 25.

To create a PBR by using the command line interface

At the command prompt, type:

- add ns pbr <name> <action> [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] [-nextHop <nextHopVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-state (ENABLED | DISABLED)]
- show ns pbr

Example

```
> add ns pbr pbr1  allow -srcip 10.102.37.252 -destip 10.10.10.2 -nexthop 10.102.29.77
 Done
```

To modify the priority of a PBR by using the command line interface

At the command prompt, type the following commands to modify the priority and verify the configuration:

- set ns pbr <name> [-action (ALLOW | DENY)] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] [-nextHop <nextHopVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-state (ENABLED | DISABLED)]
- show ns pbr [<name>]

Example

```
> set ns pbr pbr1 -priority 23
 Done
```

To remove one or all PBRs by using the command line interface

At the command prompt, type one of the following commands:

- rm ns pbr <name>
- clear ns pbrs

Example

```
> rm ns pbr pbr1
 Done
> clear ns PBRs
 Done
```

To create a PBR by using the configuration utility

Navigate to System > Network > PBRs, on the PBRs tab, add a new PBR, or edit an existing PBR.

To remove one or all PBRs by using the configuration utility

Navigate to System > Network > PBRs, on the PBRs tab, delete the PBR.

Applying a PBR

Updated: 2013-08-30

You must apply a PBR to activate it. The following procedure reapplies all PBRs that you have not disabled. The PBRs constitute a memory tree (lookup table). For example, if you create 10 PBRs (p1 - p10), and then you create another PBR (p11) and apply it, all of the PBRs (p1 - p11) are freshly applied and a new lookup table is created. If a session has a DENY PBR related to it, the session is destroyed.

You must apply this procedure after every modification you make to any PBR. For example, you must follow this procedure after disabling a PBR.

Note: PBRs created on the NetScaler appliance do not work until they are applied.

To apply a PBR by using the command line interface

At the command prompt, type:

apply ns PBRs

To apply a PBR by using the configuration utility

1. Navigate to System > Network > PBRs.
2. On the PBRs tab, select the PBR, in the Action list, select Apply.

Enabling or Disabling PBRs

Updated: 2013-08-30

By default, the PBRs are enabled. This means that when PBRs are applied, the NetScaler appliance automatically compares incoming packets against the configured PBRs. If a PBR is not required in the lookup table, but it needs to be retained in the configuration, it must be disabled before the PBRs are applied. After the PBRs are applied, the NetScaler does not compare incoming packets against disabled PBRs.

To enable or disable a PBR by using the command line interface

At the command prompt, type one of the following commands:

- enable ns pbr <name>
- disable ns pbr <name>

Examples

```
> enable ns PBR pbr1
 Done
> show ns PBR pbr1
1)     Name: pbr1
       Action: ALLOW                  Hits: 0
       srcIP = 10.102.37.252
       destIP = 10.10.10.2
       srcMac:                     Protocol:
       Vlan:                      Interface:
       Active Status: ENABLED           Applied Status: APPLIED
       Priority: 10
       NextHop: 10.102.29.77

 Done

> disable ns PBR pbr1
Warning: PBR modified, use 'apply pbrs' to commit this operation
> apply pbrs
 Done
> show ns PBR pbr1
1)     Name: pbr1
       Action: ALLOW                  Hits: 0
       srcIP = 10.102.37.252
```



```
    destIP = 10.10.10.2
    srcMac:                 Protocol:
    Vlan:                   Interface:
    Active Status: DISABLED         Applied Status: NOTAPPLIED
    Priority: 10
    NextHop: 10.102.29.77
Done
```

To enable or disable a PBR by using the configuration utility

1. Navigate to System > Network > PBRs.
2. On the PBRs tab, select the PBR, in the Action list, select Enable or Disable.

Renumbering PBRs

Updated: 2013-08-30

You can automatically renumber the PBRs to set their priorities to multiples of 10.

To renumber PBRs by using the command line interface

At the command prompt, type:

```
renumber ns pbrs
```

To renumber PBRs by using the configuration utility

Navigate to System > Network > PBRs, on the PBRs tab, in the Action list, select Renumber Priority (s).

Use Case - PBR with Multiple Hops

Updated: 2013-08-30

Consider a scenario in which two PBRs, PBR1 and PBR2, are configured on NetScaler appliance NS1. PBR1 routes all the outgoing packets, with source IP address as 10.102.29.30, to next hop router R1. PBR2 routes all the outgoing packets, with source IP address as 10.102.29.90, to next hop router R2. R3 is another next hop router connected to NS1.

If router R1 fails, all the outgoing packets that matched against PBR1 are dropped. To avoid this situation, you can specify a link load balancing (LLB) virtual server in the next hop field while creating or modifying a PBR. Multiple next hops are bound to the LLB virtual server as services (for example R1, R2, and R3). Now, if R1 fails, all the packets that matched against PBR1 are routed to R2 or R3 as determined by the LB method configured on the LLB virtual server.

The NetScaler appliance throws an error if you attempt to create a PBR with an LLB virtual server as the next hop in the following cases:

- Adding another PBR with the same LLB virtual server.
- Specifying a nonexistent LLB virtual server.
- Specifying an LLB virtual server for which the bound services are not next hops.
- Specifying an LLB virtual server for which the LB method is not set to one of the following:
  - LEASTPACKETS
  - LEASTBANDWIDTH
  - DESTIPHASH
  - SOURCEIPHASH

- WEIGHTDRR
- SRCIPDESTIP_HASH
- LTRM
- CUSTOM LOAD

- Specifying an LLB virtual server for which the LB persistence type is not set to one of the following:
  - DESTIP
  - SOURCEIP
  - SRCDSTIP

The following table lists the names and values of the entities configured on the NetScaler appliance:

Table 1. Sample Values for Creating Entities

Entity Type	Name	IP Address
Link load balancing virtual server	LLB1	NA
Services (next hops)	Router1	1.1.1.254
	Router2	2.2.2.254
	Router3	3.3.3.254
PBRs	PBR1	NA
	PBR2	NA

To implement the configuration described above, you need to:

1. Create services Router1, Router2, and Router3 that represent next hop routers R1, R2, and R3.
2. Create link load balancing virtual server LLB1 and bind services Router1, Router2, and Router3 to it.
3. Create PBRs PBR1 and PBR2, with next hop fields set as LLB1 and 2.2.2.254 (IP address of the router R2), respectively.

To create a service by using the command line interface

At the command prompt, type:

- add service <name> <IP> <serviceType> <port>
- show service <name>

Example

```
> add service Router1 1.1.1.254 ANY *
 Done
> add service Router2 2.2.2.254 ANY *
 Done
> add service Router3 3.3.3.254 ANY *
```

```
Done
```

To create a service by using the configuration utility

Navigate to Traffic Management > Load Balancing > Services, and create a service.

To create a link load balancing virtual server and bind a service by using the command line interface

At the command prompt, type:

- add lb vserver <name> <serviceType>
- bind lb vserver < name> <serviceName>
- show lb vserver < name>

Example

```
> add lb vserver LLB1 ANY
 Done
> bind lb vserver LLB1 Router1 Router2 Router3
 Done
```

To create a link load balancing virtual server and bind a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers, and create a virtual server for link load balancing. Specify **ANY** in the **Protocol** field.
   Note: Make sure that **Directly Addressable** is unchecked.
2. Under the **Services** tab, in the **Active** column, select the check box for the service that you want to bind to the virtual server.

To create a PBR by using the command line interface

At the command prompt, type:

- add ns pbr <name> <action> [-srcIP [<operator>] <srcIPVal>] [-nextHop <nextHopVal>]
- show ns pbr

Example

```
> add pbr PBR1 ALLOW -srcIP 10.102.29.30 -nextHop LLB1
 Done
> add pbr PBR2 ALLOW -srcIP 10.102.29.90 -nextHop 2.2.2.254
 Done
```

To create a PBR by using the configuration utility

Navigate to System > Network > PBRs, on the PBRs tab, add a new PBR.

Configuring a Policy-Based Routes (PBR6) for IPv6 Traffic

Mar 20, 2012

Configuring PBR6s involves the following tasks:

- Create a PBR6.
- Apply PBR6s.
- (Optional) Disable or enable a PBR6.
- (Optional) Renumber the priority of the PBR6.

Creating or Modifying a PBR6

Updated: 2013-09-06

You cannot create two PBR6s with the same parameters. If you attempt to create a duplicate, an error message appears.

You can configure the priority of a PBR6. The priority (an integer value) defines the order in which the NetScaler appliance evaluates PBR6s. When you create a PBR6 without specifying a priority, the NetScaler automatically assigns a priority that is a multiple of 10.

If a packet matches the condition defined by the PBR6, the NetScaler performs an action. If the packet does not match the condition defined by the PBR6, the NetScaler compares the packet against the PBR6 with the next highest priority.

To create a PBR6 by using the command line interface

At the command prompt, type:

- add ns pbr6 <name> <action> [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> |-protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-state (ENABLED | DISABLED)] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-nextHop <nextHopVal>] [-nextHopVlan <positive_integer>]
- show ns pbr

To modify or remove a PBR6 by using the command line interface

To modify a PBR6, type the set pbr6 <name> command and the parameters to be changed, with their new values.

To remove one or all PBR6s by using the command line interface

At the command prompt, type one of the following commands:

- rm ns pbr6 <name>
- clear ns pbr6

To create or modify a PBR6 by using the configuration utility

Navigate to System > Network > PBRs and, on the PBR6s tab, add a new PBR6, or edit an existing PBR6.

To remove one or all PBR6s by using the configuration utility

Navigate to System > Network > PBRs and, on the PBR6s tab, delete the PBR6.

Applying PBR6s

Updated: 2013-08-30

You must apply a PBR6 to activate it. The following procedure reapplies all PBR6s that you have not disabled. The PBR6s constitute a memory tree (lookup table). For example, if you create 10 PBR6s (p6_1 - p6_10), and then you create another PBR6 (p6_11) and apply it, all of the PBR6s (p6_1 - p6_11) are freshly applied and a new lookup table is created. If a session has a DENY PBR6 related to it, the session is destroyed.

You must apply this procedure after every modification you make to any PBR6. For example, you must follow this procedure after disabling a PBR6.

Note: PBR6s created on the NetScaler appliance do not work until they are applied.

To apply PBR6s by using the command line interface

At the command prompt, type:

apply ns PBR6

To apply PBR6s by using the configuration utility

1. Navigate to System > Network > PBRs.
2. On the PBR6s tab, select the PBR6, in the Action list, select Apply.

Enabling or Disabling a PBR6

Updated: 2013-08-30

By default, the PBR6s are enabled. This means that when PBR6s are applied, the NetScaler appliance automatically compares outgoing IPv6 packets against the configured PBR6s. If a PBR6 is not required in the lookup table, but it needs to be retained in the configuration, it must be disabled before the PBR6s are applied. After the PBR6s are applied, the NetScaler does not compare incoming packets against disabled PBR6s.

To enable or disable a PBR6 by using the command line interface

At the command prompt, type one of the following commands:

- enable ns pbr <name>
- disable ns pbr <name>

To enable or disable a PBR6 by using the configuration utility

1. Navigate to System > Network > PBRs.
2. On the PBR6s tab, select the PBR6, in the Action list, select Enable or Disable.

Renumbering PBR6s

Updated: 2013-08-30

You can automatically renumber the PBR6s to set their priorities to multiples of 10.

To renumber PBR6s by using the command line interface

At the command prompt, type:

renumber ns pbr6

To renumber PBR6s by using the configuration utility

Navigate to System > Network > PBRs, on the PBR6s tab, in the Action list, select Renumber Priority (s).

Troubleshooting Routing Issues

May 11, 2012

To make your troubleshooting process as efficient as possible, begin by gathering information about your network. You need to obtain the following information about the NetScaler appliance and other systems in the Network:

- Complete Topology diagram, including interface connectivity and intermediate switch details.
- Running Configuration. You can use the show running command to get the running configuration for ns.conf and ZebOS.conf.
- Output of the History command, to determine whether any configuration changes were made when the issue arose.
- Output of the Top and ps -ax commands, to determine whether any routing daemon is over utilizing the CPU or is misbehaving.
- Any routing related core files in /var/core - nsm, bgpd, ospfd, or ripd. Check the time stamp to see if they are relevant.
- dr_error.log and dr_info.log files from /var/log.
- Output of the date command and time details for all relevant systems. Print dates across all devices one after another, so that the times on the log messages can be correlated with various events.
- Relevant ns.log, newnslog files.
- Configuration files, log files and command history details from upstream and downstream routers.

This document includes the following information:

- Generic Routing FAQs
- Troubleshooting OSPF-Specific Issues

Generic Routing FAQs

Mar 20, 2012

Users typically have the following questions about how to troubleshoot generic routing issues:

- How do I save the config files?

  The write command from VTYSH saves only ZebOS.conf. Run the save ns config command from NetScaler CLI to save both ns.conf and ZebOS.conf files.
- If I have configured both a static default route and a dynamically learned default route, which is the preferred default route?

  The dynamically learned route is the preferred default route. This behavior is unique to default routes. However, in case of the Network Services Module (NSM), unless the administrative distances are modified, a statically configured route in the RIB is preferred over a dynamic route. The route that is downloaded to the NSM FIB is the static route.
- How do I block the advertisement of default routes?

  After release 7.0, the default route is not injected into ZebOS.

  However, if you are working with 7.0 or an earlier release, you must apply a suitable route map with the

  However, if you are working with 7.0 or an earlier release, you must apply a suitable route map with the redistribute kernel command for each protocol to block default route advertisement. For example:

```
ns(config)#access-list 1 deny 0.0.0.0
ns(config)#access-list 2 permit any
ns(config)#route-map redist-kernel permit 5
ns(config-route-map)#match ip address 1
ns(config)#route-map redist-kernel permit 10
ns(config-route-map)#match ip address 2
ns(config-route-map)#q
ns(config)#router ospf 1
ns(config-router)#redistribute kernel route-map redist-kernel
ns(config-router)#q
ns(config)#q
ns#show route-map
route-map redist-kernel, permit, sequence 5
  Match clauses:
    ip address 1
  Set clauses:
route-map redist-kernel, permit, sequence 10
  Match clauses:
    ip address 2
  Set clauses:
ns#show access-list
Standard IP access list 1
```

```
    deny 0.0.0.0
Standard IP access list 2
    permit any
ns#
```

- How do I view the debug output of networking daemons?

  You can write debugging output from networking daemons to a file by entering the following log file command from the global configuration view in VTYSH:

  ```
  ns(config)#log file /var/ZebOS.log
  ```

  With release 8.1, you can direct debug output to the console by entering the terminal monitor command from VTYSH user view:

  ```
  ns#terminal monitor
  ```

- How do I collect cores of running daemons?

  You can use the gcore utility to collect cores of running daemons for processing by gdb. This might be helpful in debugging misbehaving daemons without bringing the whole routing operation to a standstill.

  ```
  gcore [-s] [-c core] [executable] pid
  ```

  The -s option temporarily stops the daemon while gathering the core image. This is a recommended option, because it guarantees that the resulting image shows the core in a consistent state.

  ```
  root@ns#gcore -s -c nsm.core /netscaler/nsm 342
  ```

- How do I run a batch of ZebOS commands?

  You can run a batch of ZebOS commands from a file by entering the VTYSH -f <file-name> command. This does not replace the running configuration, but appends to it. However, by including commands to delete the existing configuration in the batch file and then add those for the new, desired configuration, you can use this mechanism to replace a specific configuration:

  ```
  !
  router bgp 234
  network 1.1.1.1 255.255.255.0
  !
  route-map bgp-out2 permit 10
   set metric 9900
   set community 8602:300
  !
  ```

Troubleshooting OSPF-Specific Issues

Mar 20, 2012

Before you start debugging any OSPF specific issue, you must collect information from the NetScaler appliance and all systems in the affected LAN, including upstream and downstream routers. To begin, enter the following commands:

1. show interface from both nscli and VTYSH
2. show ip ospf interface
3. show ip ospf neighbor detail
4. show ip route
5. show ip ospf route
6. show ip ospf database summary
    1. If there are only few LSAs in the database, then enter show ip ospf database router, show ip ospf database A. network, show ip ospf database external, and other commands to get the full details of LSAs.
    2. If there are a large number of LSAs in the database, enter the show ip ospf database self-originated command.
7. show ip ospf
8. show ns ip. This ensures that the details of all VIPs of interest are included.
9. Get the logs from peering devices and run the following command:

    gcore -s -c xyz.core /netscaler/ospfd <pid>

Note: The gcore command is non-disruptive.

Collect additional information from the NetScaler as follows:

1. Enable logging of error messages by entering the following command from the global configuration view in VTYSH:

    ns(config)#log file /var/ospf.log

2. Enable debugging ospf events and log them by using the following command:

    ns(config)#log file /var/ospf.log

    Enable debug ospf lsa packet only if the number of LSAs in the database is relatively small (< 500).

Internet Protocol version 6 (IPv6)

Mar 20, 2012

A NetScaler appliance supports both server-side and client-side IPv6 and can therefore function as an IPv6 node. It can accept connections from IPv6 nodes (both hosts and routers) and from IPv4 nodes, and can perform Protocol Translation (RFC 2765) before sending traffic to the services. You have to license the IPv6 feature before you can implement it.

The following table lists some of the IPv6 features that the NetScaler appliance supports.

Table 1. Some Supported IPv6 Features

IPv6 features
IPv6 addresses for SNIPs (NSIP6, VIP6, and SNIP6)
Neighbor Discovery (Address Resolution, Duplicated Address Detection, Neighbor Unreachability Detection, Router Discovery)
Management Applications (ping6, telnet6, ssh6)
Static Routing and Dynamic routing (OSPF)
Port Based VLANs
Access Control Lists for IPv6 addresses (ACL6)
IPv6 Protocols (TCP6, UDP6, ICMP6)
Server Side Support (IPv6 addresses for vservers, services)
USIP (Use source IP) and DSR (Direct Server Return) for IPv6
SNMP and CVPN for IPv6
HA with native IPv6 node address
IPv6 addresses for MIPs
Path-MTU discovery for IPv6

The following table lists NetScaler components that support IPv6 addresses and provides references to the topics that

document the components.

Table 2. NetScaler Components That Support IPv6 Addresses and the Corresponding Documentation

NetScaler component	Topic that documents IPv6 support
Network	Adding, Customizing, Removing, Removing all, and Viewing routes.
SSL Offload	Creating IPv6 vservers for SSL Offload
SSL Offload	Specifying IPv6 SSL Offload Monitors
SSL Offload	Creating IPv6 SSL Offload Servers
Load Balancing	Creating IPv6 vservers for Load Balancing
Load Balancing	Specifying IPv6 Load Balancing Monitors
Load Balancing	Creating IPv6 Load Balancing Servers
DNS	Creating AAAA Records

You can configure IPv6 support for the above features after implementing the IPv6 feature on your NetScaler appliance. You can configure both tagged and prefix-based VLANs for IPv6. You can also map IPv4 addresses to IPv6 addresses.

Implementing IPv6 Support

IPv6 support is a licensed feature, which you have to enable before you can use or configure it. If IPv6 is disabled, the NetScaler does not process IPv6 packets. It displays the following warning when you run an unsupported command:

"Warning: Feature(s) not enabled [IPv6PT]"

The following message appears if you attempt to run IPv6 commands without the appropriate license:

"ERROR: Feature(s) not licensed"

After licensing the feature, use either of the following procedures to enable or disable IPv6.

To enable or disable IPv6 by using the command line interface

At the command prompt, type one of the following commands:

- enable ns feature ipv6pt
- disable ns feature ipv6pt

To enable or disable IPv6 by using the configuration utility

1. Navigate to System > Settings, in the Modes and Features group, click Configure Advanced Features.
2. Select or clear the IPv6 Protocol Translation option.

VLAN Support

Updated: 2013-08-30

If you need to send broadcast or multicast packets without identifying the VLAN (for example, during DAD for NSIP, or ND6 for the next hop of the route), you can configure the NetScaler appliance to send the packet on all the interfaces with appropriate tagging. The VLAN is identified by ND6, and a data packet is sent only on the VLAN.

For more information about ND6 and VLANs, see "Configuring Neighbor Discovery."

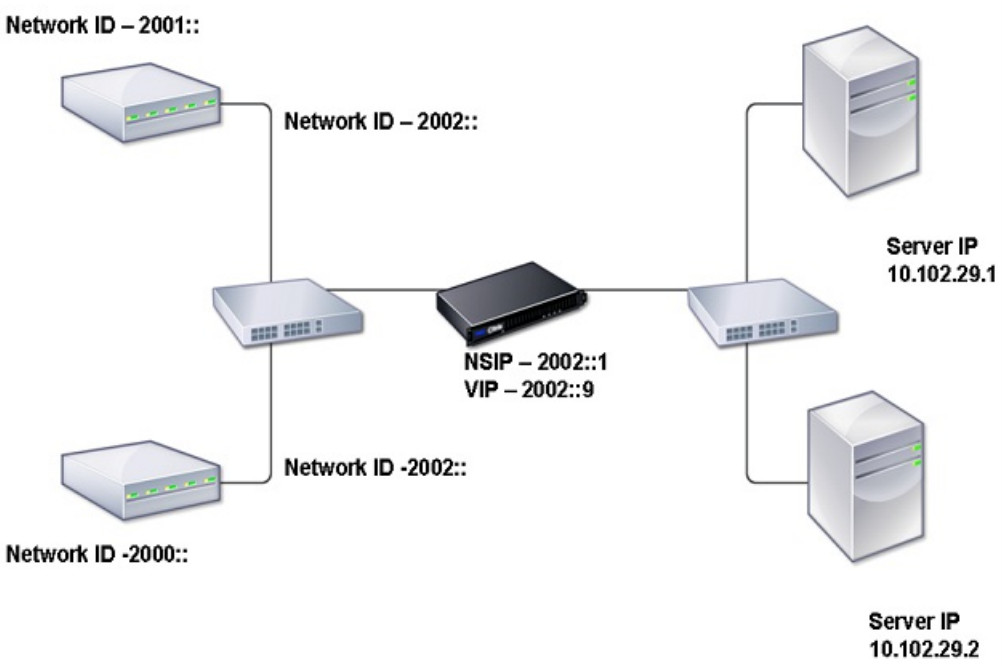
Port-based VLANs are common for IPv4 and IPv6. Prefix-based VLANs are supported for IPv6.

Simple Deployment Scenario

Updated: 2013-08-30

Following is an example of a simple load balancing set-up consisting of an IPv6 vserver and IPv4 services, as illustrated in the following topology diagram.

Figure 1. IPv6 Sample Topology



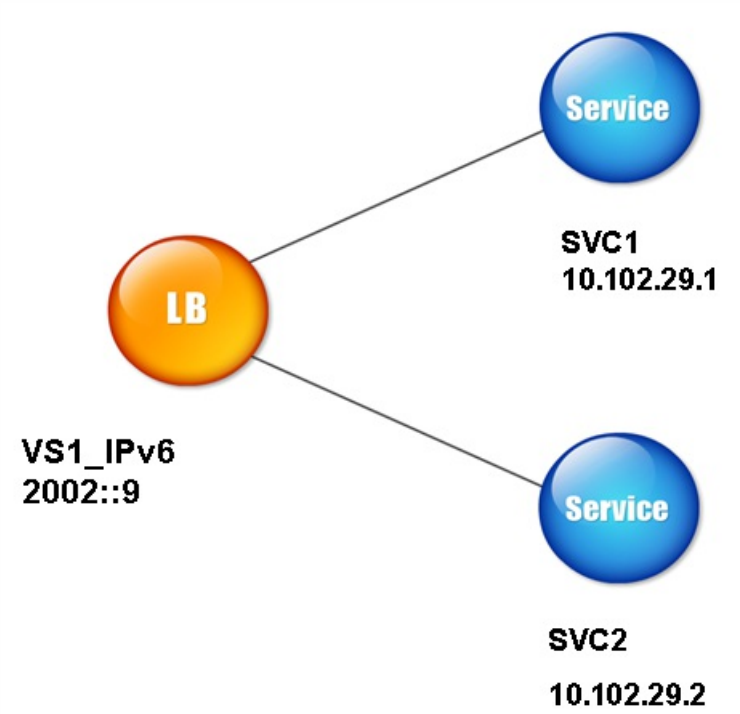
The following table summarizes the names and values of the entities that must be configured on the NetScaler.

Table 3. Sample Values for Creating Entities

Entity type	Name	Value
LB Vserver	VS1_IPv6	2002::9
Services	SVC1	10.102.29.1

Entity type	Name	Value
	SVC2	10.102.29.2

The following figure shows the entities and values of the parameters to be configured on the NetScaler.
Figure 2. IPv6 Entity Diagram



To configure this deployment scenario, you need to do the following:

1. Create an IPv6 service.
2. Create an IPv6 LB vserver.
3. Bind the services to the vserver.

To create IPv4 services by using the command line interface

At the command prompt, type:

add service <Name> <IPAddress> <Protocol> <Port>

Example

```
add service SVC1 10.102.29.1 HTTP 80
add service SVC2 10.102.29.2 HTTP 80
```

To create IPv4 services by using the configuration utility

Navigate to Traffic Management > Load Balancing > Services, click Add, and then set the following parameters:

- Service Name
- IP Address
- Protocol
- Port

To create IPv6 vserver by using the command line interface

At the command prompt, type:

add lb vserver <Name> <IPAddress> <Protocol> <Port>
Example

add lb vserver VS1_IPv6 2002::9 HTTP 80

To create IPv6 vserver by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers, click Add, and select the IPv6 check box.
2. Set the following parameters:
   - Name
   - Protocol
   - IP Address Type
   - IP Address
   - Port

To bind a service to an LB vserver by using the command line interface

At the command prompt, type:

bind lb vserver <name> <service>
Example

bind lb vserver VS1_IPv6 SVC1
The vservers receive IPv6 packets and the NetScaler performs Protocol Translation (RFC 2765) before sending traffic to the IPv4-based services.

To bind a service to an LB vserver by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers page, select the vserver for which you want to bind the service (for example, VS1_IPv6).
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box corresponding to the service that you want to bind to the vserver (for example, SVC1).
5. Click OK.
6. Repeat Steps 1-4 to bind the service (for example, SVC2 to the vserver).

Host Header Modification

Updated: 2013-08-30

When an HTTP request has an IPv6 address in the host header, and the server does not understand the IPv6 address, you must map the IPv6 address to an IPv4 address. The IPv4 address is then used in the host header of the HTTP request sent to the vserver.

To change the IPv6 address in the host header to an IPv4 address by using the command line interface

At the command prompt, type:

set ns ip6 <IPv6Address> -map <IPAddress>
Example

```
set ns ip6 2002::9 -map 200.200.200.200
```

To change the IPv6 address in the host header to an IPv4 address by using the configuration utility

1. Navigate to System > Network > IPs and, on the IPV6s tab, select the IP address for which you want to configure a mapped IP address, for example, 2002:0:0:0:0:0:0:9, and click Open.
2. In the Mapped IP text box, type the mapped IP address that you want to configure, for example, 200.200.200.200.

VIP Insertion

Updated: 2013-08-30

If an IPv6 address is sent to an IPv4-based server, the server may not understand the IP address in the HTTP header, and may generate an error. To avoid this, you can map an IPv4 address to the IPv6 VIP and enable VIP insertion.

To configure a mapped IPv6 address by using the command line interface

At the command prompt, type:

set ns ip6 <IPv6Address> -map <IPAddress>
Example

```
> set ns ip6 2002::9 -map 200.200.200.200
 Done
```

To configure a mapped IPv6 address by using the configuration utility

1. Navigate to System > Network > IPs, on the IPV6s tab, select the IP address for which you want to configure a mapped IP address, for example, 2002:0:0:0:0:0:0:9, and click Open.
2. In the Mapped IP text box, type the mapped IP address that you want to configure, for example, 200.200.200.200.

Use either of the following procedures to enable insertion of an Ipv4 VIP address and port number in the HTTP requests sent to the servers.

To enable VIP insertion by using the command line interface

At the command prompt, type:

set lb vserver <name> -insertVserverIPPort <Value>
Example

```
> set lb vserver VS1_IPv6 -insertVserverIPPort ON
 Done
```

To enable VIP insertion by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers, select the vserver that you want to enable port insertion, and click Open.
2. In the Advanced tab, under Traffic Settings, in the Vserver IP Port Insertion drop-down list box, select VIPADDR.
3. In the Vserver IP Port Insertion text box, type the vip header.

Traffic Domains

Jun 23, 2016

> **Important**
>
> Citrix recommends you to use Admin Partitioning over Traffic Domains. For more information about Admin Partitions, see Admin Partitioning page.

Traffic domains are a way to segment network traffic for different applications. You can use traffic domains to create multiple isolated environments within a NetScaler appliance. An application belonging to a specific traffic domain communicates with entities and processes traffic within that domain. The traffic belonging to one traffic domain cannot cross the boundary of another traffic domain.

Benefits of using Traffic Domains

The main benefits of using traffic domains on a NetScaler appliance are the following:

- **Use of duplicate IP addresses in a Network**. Traffic domains allow you to use duplicate IP address on the network. You can assign the same IP address or network address to multiple devices on a network, or multiple entities on a NetScaler appliance, as long as each of the duplicate address belongs to a different traffic domain.
- **Use of Duplicate entities on the NetScaler appliance**. Traffic domains also allow you to use duplicate NetScaler feature entities on the appliance. You can create entities with the same settings as long as each entity is assigned to a separate traffic domain.
  Note: Duplicate entities with same name is not supported.
- **Multitenancy**. Using traffic domains, you can provide hosting services for multiple customers by isolating each customer's type of application traffic within a defined address space on the network.

A traffic domain is uniquely identified by an identifier, which is an integer value. Each traffic domain needs a VLAN or a set of VLANs. The isolation functionality of the traffic domain depends on the VLANs bound to the traffic domain. More than one VLAN can be bound to a traffic domain, but the same VLAN cannot be a part of multiple traffic domains. Therefore, the maximum number of traffic domains that can be created depends on the number of VLANS configured on the appliance.

Default Traffic Domain

A NetScaler appliance has a preconfigured traffic domain, called the *default traffic domain*, which has an ID of 0. All factory settings and configurations are part of the default traffic domain. You can create other traffic domains and then segment traffic between the default traffic domain and each of the other traffic domains. You cannot remove the default traffic domain from the NetScaler appliance. Any feature entity that you create without setting the traffic domain ID is automatically associated with the default traffic domain.
Note: Some features and configurations are supported only in the default traffic domain. They do not work in nondefault traffic domains. For a list of the features supported in all traffic domains, see "Supported NetScaler Features in Traffic Domains ."
This section includes the following details:

- How Traffic Domains Work
- Supported NetScaler Features in Traffic Domains
- Configuring Traffic Domains

How Traffic Domains Work

As an illustration of traffic domains, consider an example in which two traffic domains, with IDs 1 and 2, are configured on NetScaler appliance NS1.

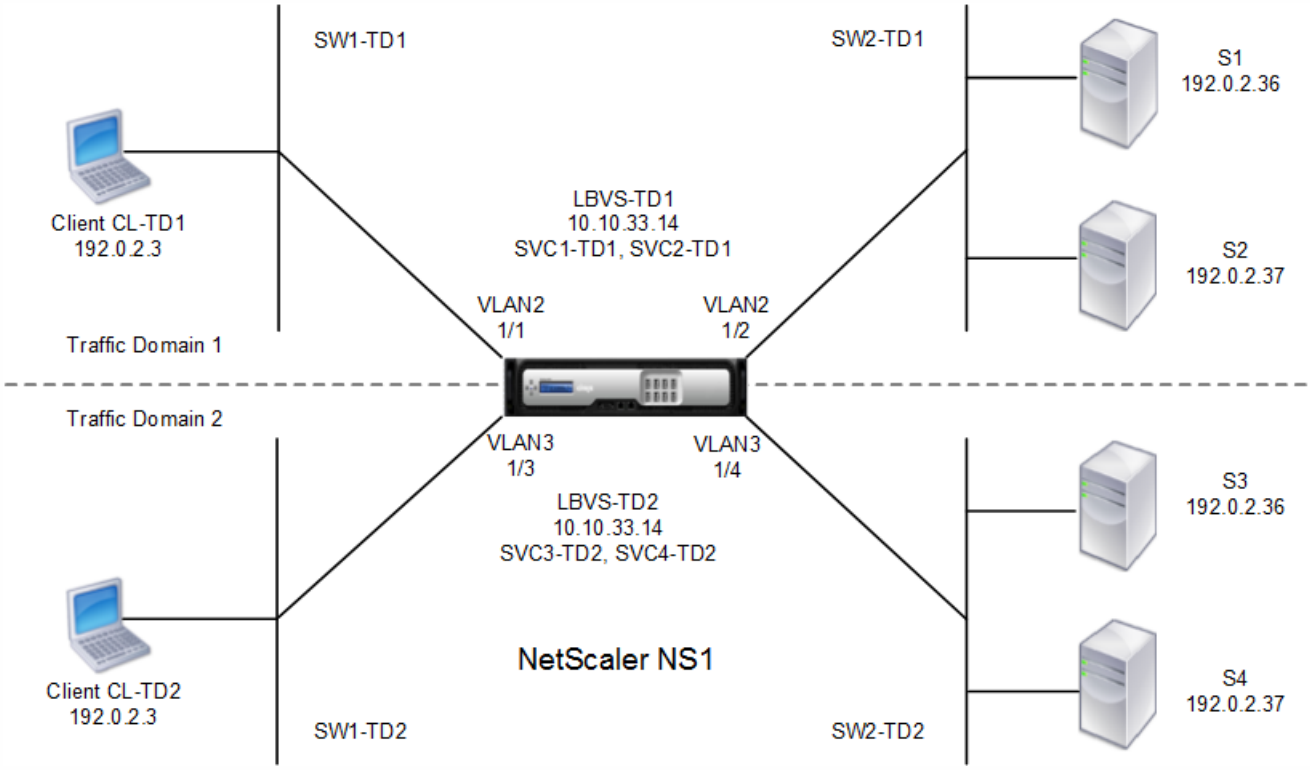
In traffic domain 1, load balancing virtual server LBVS-TD1 is configured to load balance traffic across servers S1 and S2. On the NetScaler appliance, servers S1 and S2 are represented by services SVC1-TD1 and SVC2-TD1, respectively. Servers S1 and S2 are connected to NS1 through L2 switch SW2-TD1. Client CL-TD1 is on a private network connected to NS1 through L2 switch SW1-TD1. SW1-TD1 and SW2-TD1 are connected to VLAN 2 of NS1. VLAN 2 is bound to traffic domain 1, which means that client CL-TD1 and servers S1 and S2 are part of traffic domain 1.

Similarly in traffic domain 2, load balancing virtual server LBVS-TD2 is configured to load balance traffic across S3 and S4. On the NetScaler appliance, servers S3 and S4 are represented by services SVC3-TD2 and SVC4-TD2, respectively. Servers S3 and S4 are connected to NS1 through L2 switch SW2-TD2. Client CL-TD2 is on a private network connected to NS1 through L2 switch SW1-TD2. SW1-TD2 and SW2-TD2 are connected to VLAN 3 of NS1. VLAN 3 is bound to traffic domain 2, which means that client CL-TD2 and servers S3 and S4 are part of traffic domain 2.

On the NetScaler appliance, entities LBVS-TD1 and LBVS-TD2 share the same settings, including the IP address. The same is true for SVC1-TD1 and SVC3-TD2, and for SVC2-TD1 and SVC4-TD2. This is possible because these entities are in different traffic domains.

Similarly, servers S1 and S3, S2 and S4 share the same IP address, and clients CL-TD1 and CL-TD2 each have the same IP address.

Figure 1. How traffic domains work



The following table lists the settings used in the example.

Entity	Name	Details
Settings in traffic domain 1		
VLANs bound to traffic domain 1	VLAN 2	VLAN Id: 2 Interfaces bound: 1/1, 1/2
Client connected to TD1	CL-TD1 (for reference purposes only)	IP address: 192.0.2.3
Load balancing virtual server in TD1	LBVS-TD1	IP address: 192.0.2.15
Service bound to virtual server LBVS-TD1	SVC1-TD1	IP address: 192.0.2.36
	SVC2-TD1	IP address: 192.0.2.37
SNIP	SNIP-TD1 (for reference purposes only)	IP address: 192.0.2.27
Settings in traffic domain 2		
VLAN bound to traffic domain 2	VLAN 3	VLAN Id: 3 Interfaces bound: 1/3, 1/4
Client connected to TD2	CL-TD2 (for reference purposes only)	IP address: 192.0.2.3
Load balancing virtual server in TD2	LBVS-TD2	IP address: 192.0.2.15
Service bound to virtual server LBVS-TD2	SVC3-TD2	IP address: 192.0.2.36
	SVC4-TD2	IP address: 192.0.2.37
SNIP in TD2	SNIP-TD2 (for reference purposes only)	IP address: 192.0.2.29

Following is the traffic flow in traffic domain 1:

1. Client CL-TD1 broadcasts an ARP request for the IP address of 192.0.2.15.
2. The ARP request reaches NS1 on interface 1/1, which is bound to VLAN 2. Because VLAN 2 is bound to traffic domain 1, NS1 updates the ARP table of traffic domain 1 for the IP address of client CL-TD1.
3. Because the ARP request is received on traffic domain 1, NS1 looks for an entity configured on traffic domain 1 that has an IP address of 192.0.2.15. NS1 finds that a load balancing virtual server LBVS-TD1 is configured on traffic domain 1 and has the IP address 192.0.2.15.
4. NS1 sends an ARP response with the MAC address of interface 1/1.

5. The ARP reply reaches CL-TD1. CL-TD1 updates its ARP table for the IP address of LBVS-TD1 with the MAC address of interface 1/1 of NS1.
6. Client CL-TD1 sends a request to 192.0.2.15. The request is received by LBVS-TD1 on port 1/1 of NS1.
7. LBVS-TD1's load balancing algorithm selects server S2, and NS1 opens a connection between a SNIP in traffic domain 1 (192.0.2.27) and S2.
8. S2 replies to SNIP 192.0.2.27 on NS1.
9. NS1 sends S2's reply to client CL-TD1.

Following is the traffic flow in traffic domain 2:

1. Client CL-TD2 broadcasts an ARP request for the IP address of 192.0.2.15.
2. The ARP request reaches NS1 on interface 1/3, which is bound to VLAN 3. Because VLAN 3 is bound to traffic domain 2, NS1 updates traffic-domain 2's ARP-table entry for the IP address of client CL-TD2, even though an ARP entry for the same IP address (CL-TD1) is already present in the ARP table of traffic domain 1.
3. Because the ARP request is received in traffic domain 2, NS1 searches traffic domain 2 for an entity that has an IP address of 192.0.2.15. NS1 finds that load balancing virtual server LBVS-TD2 is configured in traffic domain 2 and has the IP address 192.0.2.15. NS1 ignores LBVS-TD1 in traffic domain 1, even though it has the same IP address as LBVS-TD2.
4. NS1 sends an ARP response with the MAC address of interface 1/3.
5. The ARP reply reaches CL-TD2. CL-TD2 updates its ARP table entry for the IP address of LBVS-TD2 with the MAC address of interface 1/3 of NS1.
6. Client CL-TD2 sends a request to 192.0.2.15. The request is received by LBVS-TD2 on interface 1/3 of NS1.
7. LBVS-TD2's load balancing algorithm selects server S3, and NS1 opens a connection between a SNIP in traffic domain 2 (192.0.2.29) and S3.
8. S2 replies to SNIP 192.0.2.29 on NS1.
9. NS1 sends S2's reply to client CL-TD2.

Supported NetScaler Features in Traffic Domains

The NetScaler features in the following list are supported in all traffic domains.

- ARP table
- ND6 table
- Bridge table
- All types of IPv4 and IPv6 addresses
- IPv4 and IPv6 routes
- ACL and ACL6
- PBR & PBR6
- INAT
- RNAT
- RNAT6
- MSR
- MSR6
- Net profiles
- SNMP MIBs
- Fragmentation
- Monitors (Scriptable Monitors are not supported)
- Content Switching
- Persistency
- Service (Domain based services are not supported)
- Servicegroup (Domain based service groups are not supported)
- Policies (*)
- PING
- TRACEROUTE
- PMTU
- High Availability (connection mirroring is not supported)
- Cluster (Supported on L2 clusters. Not supported on L3 clusters)
- Cookie Persistency
- MSS
- Logging
- Priority Queuing
- Surge Protection
- HTTP DOSP (*)
- Load balancing (The following types are not supported:
  - TFTP

• Cache Redirection	• RTSP • Diameter • SIP) • NAT46 • NAT64 • DNS64 • Forwarding Session Rules • SNMP

Important

Any NetScaler feature not listed above is supported only in the default traffic domain. Traffic domains are not supported in a cluster configuration.

Note

Global Server Loading Balancing (GSLB) and ADNS features in NetScaler are not aware of Traffic Domains. If the GSLB configuration needs to be shared across all traffic domains then GSLB methods Static Proximity and Round Trip Time (RTT) do not work. As a workaround in this scenario, you can use GSLB methods other than RTT and Static Proximity. For more information, see http://support.citrix.com/article/CTX202277.

Configuring Traffic Domains

Configuring a traffic domain on the NetScaler appliance consists of the following tasks:

- **Add VLANs.** Create VLANs and bind specified interfaces to them.
- **Create a traffic domain entity and bind VLANs to it.** This involves the following two tasks:
  - Create a traffic domain entity uniquely identified by an ID, which is an integer value.
  - Bind the specified VLANs to the traffic domain entity. All the interfaces that are bound to the specified VLANs are associated with the traffic domain. More than one VLAN can be bound to a traffic domain, but a VLAN cannot be a part of multiple traffic domains.
- **Create feature entities on the traffic domain**. Create the required feature entities in the traffic domain. The CLI commands and configuration dialog boxes of all the supported features in a nondefault traffic domain include a parameter called a *traffic domain identifier* (td). When configuring a feature entity, if you want the entity to be associated with a particular traffic domain, you must specify the td. Any feature entity that you create without setting the td is automatically associated with the default traffic domain.

To give you an idea of how feature entities are associated with a traffic domain, this topic covers the procedures for configuring all the entities mentioned in the figure titled "How Traffic Domains Work."

The command line interface has two commands for these two tasks, but the configuration utility combines them in a single dialog box.

To create a VLAN and bind interfaces to it by using the command line interface

At the command prompt, type:

- add vlan <id>
- bind vlan <id> -ifnum <slot/port>
- show vlan <id>

To create a traffic domain entity and bind VLANs to it by using the command line interface

At the command prompt, type:

- add ns trafficdomain <td>
- bind ns trafficdomain <td> -vlan <id>
- show ns trafficdomain <td>

To create a service by using the command line interface

At the command prompt, type:

- add service <name> <IP> <serviceType> <port> -td <id>
- show service <name>

To create a load balancing virtual server and bind services to it by using the command line interface

At the command prompt, type:

- add lb vserver <name> <serviceType> <IPAddress> <port> -td <id>
- bind lb vserver <name> <serviceName>
- show lb vserver <name>

To create a VLAN by using the configuration utility

Navigate to System > Network > VLANs, click Add, and set the parameters.

To create a traffic domain entity by using the configuration utility

Navigate to System > Network > Traffic Domains, click Add, and in the Create Traffic Domain dialog box, set the parameters.

To create a service by using the configuration utility

Navigate to Traffic Management > Load Balancing > Services, click Add, and set the parameters.

To create a load balancing virtual server by using the configuration utility

Navigate to Traffic Management > Load Balancing > Virtual Servers, click Add, and set the parameters.

AppExpert

Jan 07, 2014

The following topics provide a conceptual reference and configuration instructions for the AppExpert and other features of the NetScaler appliance.

Action Analytics	Collects run-time statistics on the basis of pre-defined criteria. When used with policies, the feature also provides you with the infrastructure for automatic, real-time traffic optimization.
AppExpert Applications and Templates	Simplify configuration steps for the Citrix® NetScaler® appliance by using applications, application templates, NetScaler Gateway applications, and entity templates.
Entity Templates	Describes how to use entity templates to set up and configure individual NetScaler entities, such as a policy or virtual server. An entity template provides a specification and a set of defaults for the object.
AppQoE	Application level Quality of Experience (AppQoE) integrates several existing policy-based security features of the NetScaler appliance into a single integrated feature that takes advantage of a new queuing mechanism, fair queuing.
HTTP Callouts	An HTTP request that the NetScaler appliance generates and sends to an external application when certain criteria are met during policy evaluation.
Pattern Sets	Allow string matching during the evaluation of a default syntax policy.
Policies and Expressions	Rules that determine the operations that the NetScaler appliance must perform.
Rate Limiting	Defines the maximum load for a given network entity or virtual entity on the NetScaler appliance.
Responder	Bases responses on who sends the request, where it is sent from, and other criteria with security and system management implications.
Rewrite	Rewrites information in the requests or responses handled by the NetScaler appliance.
String Maps	Perform pattern matching in all NetScaler features that use the default policy syntax.

Action Analytics

Sep 16, 2013

The performance of your website or application depends on how well you optimize the delivery of the most frequently requested content. Techniques such as caching and compression help accelerate the delivery of services to clients, but you need to be able to identify the resources that are requested most frequently, and then cache or compress those resources. You can identify the most frequently used resources by aggregating real-time statistics about website or application traffic. Statistics such as how frequently a resource is accessed relative to other resources and how much bandwidth is consumed by those resources help you determine whether those resources need to be cached or compressed to improve server performance and network utilization. Statistics such as response times and the number of concurrent connections to the application help you determine whether you must enhance server-side resources.

If the website or application does not change frequently, you can use products that collect statistical data, and then manually analyze the statistics and optimize the delivery of content. However, if you do not want to perform manual optimizations, or if your website or application is dynamic in nature, you need infrastructure that can not only collect statistical data but can also automatically optimize the delivery of resources on the basis of the statistics. On the NetScaler appliance, this functionality is provided by the action analytics feature. The feature operates on a single NetScaler appliance and collects run-time statistics on the basis of criteria that you define. When used with NetScaler policies, the feature also provides you with the infrastructure that you need for automatic, real-time traffic optimization.

When configuring the action analytics feature, you specify the request attributes for which you want to collect statistical data (for example, URLs and HTTP methods) by configuring default syntax expressions in an entity called a selector. Then, you configure an identifier to configure settings such as the sampling interval and sample count. You also configure a policy that enables the appliance to evaluate traffic as specified by the selector-identifier pair. Finally, you bind the policy to a bind point to begin collecting statistics.

The appliance also provides you with a set of built-in selectors, identifiers, and responder policies that you can use to get started with the feature.

The appliance aggregates the following statistics:

- The number of requests.
- The bandwidth consumed by the requests.
- The response time.
- The number of concurrent connections.

You can configure the feature to perform run-time sorting of the records on an attribute of your choice. You can view the statistical data by using either the command-line interface or the Stream Sessions tool in the configuration utility.

Configuring a Selector

Sep 16, 2013

A selector is a filter for identifying requests. It consists of up to five individual default syntax expressions that identify request attributes such as the client IP address and the URL in the request. Each expression is a non-compound default syntax expression and is considered to be in an AND relationship with the other expressions. Following are some examples of selector expressions:

- HTTP.REQ.URL
- CLIENT.IP.SRC
- HTTP.RES.BODY(1000).AFTER_STR(\"<string>\").BEFORE_STR(\"<string>\")"
- CLIENT.IP.SRC.SUBNET(24)

Selectors are used in rate limiting and action analytics configurations. A selector is optional in a rate limiting configuration, but is required in a action analytics configuration.

The order in which you specify parameters is significant. For example, if you configure an IP address and a domain (in that order) in one selector, and then specify the domain and the IP address (in the reverse order) in another selector, the NetScaler considers these values to be unique. This can lead to the same transaction being counted twice. Also, if multiple policies invoke the same selector, the NetScaler, again, can count the same transaction more than once.

If you modify an expression in a selector, you may get an error if any policy that invokes it is bound to a new policy label or bind point. For example, suppose that you create a selector named myLimitSelector1, invoke it from myLimitID1, and invoke the identifier from a DNS policy named dnsRateLimit1. If you change the expression in myLimitSelector1, you might receive an error when binding dnsRateLimit1 to a new bind point. The workaround is to modify these expressions before creating the policies that invoke them.

The NetScaler appliance provides the following built-in selectors for some of the most common use cases:

Table 1. Built-in Selectors

Selector	Selector Expressions
Top_URL	HTTP.REQ.URL
Top_CLIENTS	CLIENT.IP.SRC
Top_URL_CLIENTS_LBVSERVER	1. HTTP.REQ.URL 2. CLIENT.IP.SRC 3. HTTP.REQ.LB_VSERVER.NAME
Top_URL_CLIENTS_CSVSERVER	1. HTTP.REQ.URL 2. CLIENT.IP.SRC 3. HTTP.REQ.CS_VSERVER.NAME
Top_MSSQL_QUERY_DB_LBVSERVER	1. MSSQL.REQ.QUERY.TEXT 2. MSSQL.REQ.LB_VSERVER.NAME

Selector	Selector Expressions
Top_MYSQL_QUERY_DB_LBVSERVER	1. MYSQL.REQ.QUERY.TEXT 2. MYSQL.REQ.LB_VSERVER.NAME

You can also configure a selector with expressions that identify the request attributes of your choice. For example, you might want to create a record for a request that arrives with a specific header. To evaluate the header, you can add HTTP.REQ.HEADER("<header_name>") to the selector that you intend to use.

To configure a selector by using the command line interface

At the command prompt, type the following commands to configure a selector and verify the configuration:

- add stream selector <name> <rule> ...
- show stream selector

Example

```
> add stream selector myselector HTTP.REQ.URL CLIENT.IP.SRC
 Done
> show stream selector myselector
 Name: myselector
 Expressions:
  1) HTTP.REQ.URL
  2) CLIENT.IP.SRC
 Done
>
```

To modify or remove a selector by using the command line interface

- To modify a selector, type the set stream selector command, the name of the selector, and the rule parameter with the expressions. Enter the existing expressions that you want to retain, along with the new expressions that you want to add.
- To remove a selector, type the rm stream selector command and the name of the selector.

To configure a selector by using the configuration utility

1. Navigate to AppExpert > Action Analytics > Selectors.
2. In the details pane, do one of the following:
   - To create a selector, click Add.
   - To modify a selector, select the selector, and then click Open.
3. In the Create Limit Selector or Configure Limit Selector dialog box, set one or more of the following parameters:
   - Name
   - Expressions
     To add the expression to the selector configuration, click Add. To remove an expression from the selector configuration, in the Expression box, select the expression, and then click Remove.
     Note: In the Expressions box, enter a valid parameter. For example, enter HTTP. Then, enter a period after this parameter. A drop-down menu appears. The contents of this menu provide the keywords that can follow the initial keyword that you entered. To select the next keyword in this expression prefix, double-click the selection in the drop-down menu. The Expressions text box displays both the first and second keywords for the expression prefix, for

example, HTTP.REQ. Continue adding expression components until the complete expression is formed.

4. Click Add.
5. Continue adding up to five non-compound expressions.
6. Click Create or OK.

Configuring a Stream Identifier

Aug 30, 2013

You configure a stream identifier to specify parameters for collecting statistical data from requests identified by a given selector. An identifier specifies the selector to be used, the statistics collection interval, the sample count, and the field on which the records are to be sorted.

The NetScaler appliance includes the following built-in stream identifiers for common use cases. All the built-in identifiers specify a sample count of 1 and an interval of 1 minute. Additionally, they sort the data on the REQUESTS attribute. They differ only in being associated with different built-in selectors. Each built-in identifier is associated with a built-in selector of the same name (for example, the built in identifier Top_URL is associated with the built-in selector Top_URL). Following are the built-in identifiers:

- Top_URL
- Top_CLIENTS
- Top_URL_CLIENTS_LBVSERVER
- Top_URL_CLIENTS_CSVSERVER
- Top_MSSQL_QUERY_DB_LBVSERVER
- Top_MYSQL_QUERY_DB_LBVSERVER

For more information about the built-in selectors, see "Configuring a Selector."

Note: The maximum length for storing string results of selectors (for example, HTTP.REQ.URL) is 60 characters. If the string (for example, URL) is 1000 characters long, of which 50 characters are enough to uniquely identify a string, use an expression to extract only the required 50 characters.
You cannot modify a built-in identifier's configuration. However, you can create an identifier with a configuration of your choice.

To configure a stream identifier by using the command line interface

At the command prompt, type the following commands to configure a stream identifier and verify the configuration:

- add stream identifier <name> <selectorName> [-interval <positive_integer>] [-SampleCount <positive_integer>] [-sort <sort>]
- show stream identifier <name>

Example

```
> add stream identifier myidentifier Top_URL -interval 10 -sampleCount 100
 Done
```

To configure a stream identifier by using the configuration utility

1. Navigate to AppExpert > Action Analytics > Stream Identifiers.
2. In the details pane, do one of the following:
   - To create a stream identifier, click Add.
   - To modify a stream identifier, select the identifier, and then click Open.
3. In the Configure Stream Identifier dialog box, set one or more of the following parameters:
   - Name
   - Selector
   - Interval

- Sample Count
- Sort

4. Click Create or OK, and then click Close.

Viewing Statistics

Aug 30, 2013

You can view the collected statistics in tabular format in the command-line interface and in graphical format in the configuration utility.

The following table describes the collected statistics:

Table 1. Statistical Information Displayed for a Stream Identifier

Statistics	Column name in the output of the stat stream identifier <identifier name> command	Description
Number of requests	Req	The number of requests for which records were created in the last <interval> number of minutes.
Bandwidth consumed	BandW	The total bandwidth consumed by the requests that were received in the last <interval> number of minutes. The total bandwidth of a request is the bandwidth consumed by the request and its response. The value is rounded off to the next higher or next lower integer value. Consequently, it might differ slightly from the expected value. For example, if a request's total bandwidth consumption is 2.2 KB, one instance of the request might be shown as having consumed 2 KB and two instances might be shown as having consumed 4 KB, but three instances might be shown as having consumed 7 KB.
Response time	RspTime	The average response time for all the requests received in the last <interval> number of minutes.
Concurrent connections	Conn	The total number of concurrent connections that are currently open.

To view the statistical data collected for a stream identifier by using the command line

At the command prompt, type:

stat stream identifier <name> [<pattern> ...] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-sortBy <sortBy> [<sortOrder>]

Examples

Example 1 sorts the output on the BandW column, in the descending order. Example 2 sorts the output in Example 1, on the Req column, and in the ascending order

Example 1

```
> stat stream identifier myidentifier -sortBy BandW Descending -fullValues
Stream Session statistics
            Req          BandW
User1        508         125924
User2       5020          12692
User3       2025           4316

          RspTime          Conn
User1       5694             0
User2        109             0
User3          3             0
 Done
```

Example 2

```
> stat stream identifier myidentifier -sortBy Req Ascending –fullValues
Stream Session statistics
            Req          BandW
User1        508         125924
User3       2025           4316
User2       5020          12692

          RspTime          Conn
User1       5694             0
User3          3             0
User2        109             0
 Done
```

To view the statistical data collected for a stream identifier by using the configuration utility

1. Navigate to AppExpert > Action Analytics > Stream Identifiers.
2. Select the stream identifier whose sessions you want to view, and then click Stream Sessions. For information about how you can group the output on the basis of the values collected for various selector expressions, see "Grouping Records on Attribute Values."

Grouping Records on Attribute Values

Sep 30, 2013

Statistical information such as the number of times a particular URL has been accessed overall and per client, and the total number of GET and POST requests per client can provide valuable insights into whether any of your resources need to be expanded to meet the demand or be optimized for delivery. To obtain such statistics, you must use an appropriate set of selector expressions, and then use the pattern parameter in the stat stream identifier command. The grouping is based on the pattern that is specified in the command. Grouping can be performed concurrently on the values of multiple expressions.

In the command-line interface, you can group the output by using patterns of your choice. In the configuration utility, the pattern depends on the choices you make when drilling down through the values of various selector expressions. For example, consider a selector that has the expressions HTTP.REQ.URL, CLIENT.IP.SRC, and HTTP.REQ.LB_VSERVER.NAME, in that order. The statistics home page displays icons for each of these expressions. If you click the icon for CLIENT.IP.SRC, the output is based on the patterns * ? *. The output displays statistics for each client IP address. If you click an IP address, the output is based on the patterns * <IP address> ? and ? <IP address> * where <IP address> is the IP address you selected. In the resulting output, if you click a URL, the pattern used is <URL> <IP address> ?.

To group the records on the values of selector expressions by using the command line interface

At the command prompt, enter the following command to group the records on the basis of a selector expression:

stat stream identifier <name> [<pattern> ...]

Examples

Each example uses a different pattern to demonstrate the effect of the pattern on the output of the stat stream identifier command. The selector expressions are HTTP.REQ.URL and HTTP.REQ.HEADER("UserHeader"), in that order. The requests contain a custom header whose name is UserHeader. Note that in the examples, a given statistical value changes as determined by the grouping, but the sum total of the values for a given field remains the same.

Example 1

In the following command, the pattern used is ? ?. The appliance groups the output on the values collected for both selector expressions. The row headers consist of the expression values separated by a question mark (?). The row with the header /mysite/mypage1.html?Ed displays statistics for requests made by user Ed for the URL /mysite/mypage1.html.

```
> stat stream identifier myidentifier ? ? -fullValues
Stream Session statistics
                          Req          BandW
/mysite/mypage2.html?Grace          1           2553
/mysite/mypage1.html?Grace          2             4
/mysite/mypage1.html?Ed             8            16
/mysite/mypage2.html?Joe            1          2554
/mysite/mypage1.html?Joe            5            10
/mysite/?Joe                    1           4

                       RspTime          Conn
```

```
/mysite/mypage2.html?Grace              0              0
/mysite/mypage1.html?Grace              0              0
/mysite/mypage1.html?Ed                 0              0
/mysite/mypage2.html?Joe                0              0
/mysite/mypage1.html?Joe                0              0
/mysite/?Joe                        6              0
 Done
```

Example 2

In the following command, the pattern used is * ?. The appliance groups the output on the values accumulated for the second expression HTTP.REQ.HEADER("UserHeader"). The rows display statistics for all requests made by users Grace, Ed, and Joe.

```
> stat stream identifier myidentifier * ?
Stream Session statistics
          Req   BandW  RspTime    Conn
Grace       3    2557      0      0
Ed          8      16      0      0
Joe         7    2568      6      0
 Done
```

Example 3

In the following command, the pattern used is ? *, which is the default pattern. The output is grouped on the values collected for the first selector expression. Each row displays statistics for one URL.

```
> stat stream identifier myidentifier ? * -fullValues
Stream Session statistics
                        Req           BandW
/mysite/mypage2.html              2            5107
/mysite/mypage1.html             15              30
/mysite/                      1             4

                      RspTime           Conn
/mysite/mypage2.html              0             0
/mysite/mypage1.html              0             0
/mysite/                      6             0
 Done
```

Example 4

In the following command, the pattern used is * *. The appliance displays one set of collective statistics for all the requests received, with no row title.

```
> stat stream identifier myidentifier * *
Stream Session statistics
          Req   BandW  RspTime    Conn
           18    5141      6      0
 Done
```

Example 5

In the following command, the pattern is /mysite/mypage1.html *. The appliance displays one set of collective statistics

for all the requests received for the URL /mysite/mypage1.html, with no row title.

```
> stat stream identifier myidentifier /mysite/mypage1.html *
Stream Session statistics
          Req    BandW  RspTime    Conn
           15      30       0       0
 Done
```

To group the records on the values of selector expressions by using the configuration utility

1. Navigate to AppExpert > Action Analytics > Stream Identifiers.
2. In the details pane, click the stream identifier for which you want to view statistics, and then click Stream Sessions.
3. On the Home page, click the icon for the stream selector by which you want to group the output.
4. To return to the Home page from the statistics page for a selector expression, click Home.
5. To view statistics for the value of a given selector expression, click the value. You can repeat this step for a selector expression value in each subsequent output until you obtain the statistics you want.

Clearing a Stream Session

Aug 30, 2013

You can flush all the records that have been accumulated for a stream identifier.

To clear a stream session by using the command line interface

At the command prompt, enter the following commands to clear a stream session and verify the results:

- clear stream session <name>
- stat stream identifier <name>

Example

This example uses the stat stream identifier command first, so that a comparison can be made with the stat stream identifier command that is used for verifying the result of the clear stream session command.

```
>stat stream identifier myidentifier
Stream Session statistics
            Req    BandW  RspTime    Conn
/aed....html      2       0       0       0
/             636     303      12       0
 Done
>clear stream session myidentifier
 Done
>stat stream identifier myidentifier
 Done
```

To clear a stream session by using the configuration utility

1. Navigate to AppExpert > Action Analytics > Stream Identifiers.
2. Select the stream identifier whose sessions you want to clear, and then click Clear Sessions.

Configuring a Policy for Analyzing and Optimizing Traffic

Sep 16, 2013

To put the selector-identifier pair in your action analytics configuration into effect, you must associate the pair with the point in the traffic flow at which you want to collect statistics. You can do so by configuring a default syntax policy and referencing the stream identifier from the policy rule. You can use compression policies, caching policies, rewrite policies, application firewall policies, responder policies, and any other policies whose action is based on a Boolean expression.

The action analytics feature introduces a set of default syntax expressions and functions for collecting and evaluating data. The expression ANALYTICS.STREAM(<identifier_name>) is used for referencing the identifier that you want to use. The expression COLLECT_STATS is used to collect statistical data. Functions such as IS_TOP(<uint>) and IS_TOP_FREQUENTS(<uint>) are used for making automatic, real-time traffic optimization decisions.

- **IS_TOP(<number>).** Finds if a given object is in the top <number> of elements. For example, is the element among the top 10 elements. When multiple elements have the count, they are considered to be similar in nature. The sort function must be turned on to avoid an undef condition.
- **IS_TOP_FREQUENTS(<frequency>).** Finds if a given object is in the top <frequency> of the elements that are in the top elements. For example, is the element among the top 50% of all the top elements maintained. Elements having the same values are considered similar in nature. The sort function must be turned on to avoid an undef condition.

It is your policy configuration that determines whether the NetScaler appliance must only collect data from traffic or also perform an action. If the appliance must only collect statistical data, you can configure a policy with the rule ANALYTICS.STREAM(<identifier_name>).COLLECT_STATS and the action NOOP. The NOOP policy must be the policy with the highest priority at the bind point. This policy is sufficient if you are only collecting statistics. Traffic optimization decisions, such as what to compress or cache, must be based on manual, periodic evaluation of the statistical data.

If, in addition to collecting statistics, the appliance must also perform an action on the traffic, you must configure the gotoPriorityExpression parameter of the NOOP policy such that another policy that has the desired rule and action is evaluated subsequently. This second policy must have a rule that begins with the ANALYTICS.STREAM(<identifier_name>) prefix and a function that evaluates the data.

Following is an example of two responder policies that are configured and bound globally. The policy responder_stat_collection enables the appliance to collect statistics based on the identifier, myidentifier. The policy responder_notify evaluates the data that is collected.

Example

```
> add responder action send_notification respondwith '"You are in the Top 10 list for bandwidth consumption"'
 Done
> add responder policy responder_stat_collection' ANALYTICS.STREAM("myidentifier").COLLECT_STATS' NOOP
Done
> add responder policy responder_notify 'ANALYTICS.STREAM("myidentifier").BANDWIDTH.IS_TOP(10)' send_notification
 Done
> bind responder global responder_stat_collection 10 NEXT
 Done
> bind responder global responder_notify 20 END
 Done
```

Use Case: Limiting Bandwidth Consumption per User or Client Device

Sep 16, 2013

Your web site, application, or file hosting service has finite network and server resources available to it to serve all its users. One of the most important resources is bandwidth. Substantial bandwidth consumption by only a subset of the user base can result in network congestion and reduced resource availability to other users. To prevent network congestion, you might have to limit a client's bandwidth consumption by using temporary service denial techniques such as responding to a client request with an HTML page if it has exceeded a preconfigured bandwidth value over a fixed time period leading up to the request.

In general, you can regulate bandwidth consumption either per client device or per user. This use case demonstrates how you can limit bandwidth consumption per client to 100 MB over a time period of one hour. The use case also demonstrates how you can regulate bandwidth consumption per user to 100 MB over a time period of one hour, by using a custom header that provides the user name. In both cases, the tracking of bandwidth consumption over a moving time period of one hour is achieved by setting the interval parameter in the stream identifier to 60 minutes. The use cases also demonstrate how you can import an HTML page to send to a client that has exceeded the limit. Importing an HTML page not only simplifies the configuration of the responder action in these use cases, but also simplifies the configuration of all responder actions that need the same response.

To limit bandwidth consumption per user or client device by using the command line interface

In the command-line interface, perform the following tasks to configure action analytics for limiting a client's or user's bandwidth consumption. Each step includes sample commands and their output.

1. **Set up your load balancing configuration.** Configure load balancing virtual server mysitevip, and then configure all the services that you need. Bind the services to the virtual server. The following example creates ten services and binds the services to mysitevip.

   ```
   > add lb vserver mysitevip HTTP 192.0.2.17 80
    Done
   > add service service[1-10] 192.0.2.[240-249] HTTP 80
   service "service1" added
   service "service2" added
   service "service3" added
   .
   .
   .
   service "service10" added
    Done
   > bind lb vserver vserver1 service[1-10]
   service "service1" bound
   service "service2" bound
   service "service3" bound
   .
   .
   .
   service "service10" bound
    Done
   ```

2. **Configure the stream selector.** Configure one of the following stream selectors:
   - To limit bandwidth consumption per client, configure a stream selector that identifies the client IP address.

     ```
     > add stream selector myselector CLIENT.IP.SRC
      Done
     ```

- To limit bandwidth consumption per user on the basis of the value of a request header that provides the user name, configure a stream selector that identifies the header. In the following example, the name of the header is UserHeader.

  ```
  > add stream selector myselector HTTP.REQ.HEADER("UserHeader")
   Done
  ```

3. **Configure a stream identifier.** Configure a stream identifier that uses the stream selector. Set the interval parameter to 60 minutes.

   ```
   > add stream identifier myidentifier myselector -interval 60 -sampleCount 1 -sort BANDWIDTH
    Done
   ```

4. **Configure the responder action.** Import the HTML page that you want to send to users or clients that have exceeded the bandwidth consumption limit, and then use the page in responder action crossed_limits.

   ```
   > import responder htmlpage http://192.0.2.20:80/stdpages/wait.html crossed-limits.html
   This operation may take some time, Please wait...

    Done
   > add responder action crossed_limits respondwithhtmlpage crossed-limits.html
    Done
   ```

5. **Configure the responder policies.** Configure responder policy myrespol1 with the rule ANALYTICS.STREAM("myidentifier").COLLECT_STATS and the action NOOP. Then, configure policy myrespol2 for determining whether a client or user has crossed the 100 MB limit. The policy myrespol2 is configured with the responder action crossed_limits.

   ```
   > add responder policy myrespol1 'ANALYTICS.STREAM("myidentifier").COLLECT_STATS' NOOP
    Done
   > add responder policy myrespol2 'ANALYTICS.STREAM("myidentifier").BANDWIDTH.GT(1048576)' crossed_limits
    Done
   ```

6. **Bind the responder policies to the load balancing virtual server.** The policy myrespol1, which only collects statistical data, must have the higher priority and a GOTO expression of NEXT.

   ```
   > bind lb vserver mysitevip -policyName myrespol1 -priority 1 -gotoPriorityExpression NEXT
    Done
   > bind lb vserver mysitevip -policyName myrespol2 -priority 2 -gotoPriorityExpression END
    Done
   ```

7. **Test the configuration.** Test the configuration by sending test HTTP requests, from multiple clients or users, to the load balancing virtual server and using the stat stream identifier command to view the statistics that are collected for the specified identifier. The following output displays statistics for clients.

   ```
   > stat stream identifier myidentifier -sortBy BandW –fullValues
   Stream Session statistics
                     Req        BandW
   192.0.2.30          5000       3761
   192.0.2.31          29         2602
   192.0.2.32          25         51

                     RspTime      Conn
   192.0.2.30          2          0
   192.0.2.31          0          0
   192.0.2.32          0          0
    Done
   >
   ```

AppExpert Applications and Templates

May 21, 2015

An AppExpert application is a collection of configuration information that you set up on the Citrix NetScaler appliance for securing and optimizing traffic for a Web application, such as Microsoft SharePoint. Managing AppExpert applications is simplified by a graphical user interface (GUI) that allows you to specify application traffic subsets and a distinct set of security and optimization policies for processing each traffic subset. Additionally, it consolidates all deployment tasks in one view, so you can quickly configure target IP addresses for clients and specify host servers.

Prebuilt application templates for widely used Web applications, such as Microsoft Outlook Web Access and Microsoft SharePoint, are available on the AppExpert Templates page of the Citrix Community website at "http://community.citrix.com/display/ns/AppExpert+Templates."

Each prebuilt template provides you with an initial configuration for managing the associated Web application. You can customize prebuilt application templates for your organization. If a prebuilt application template does not suit your requirements, you can create a custom application without using a template.

Regardless of whether you use a prebuilt application template or you create a custom application, you can export the configuration to a template file. You can then share the template with other administrators or import the template to other NetScaler appliances that require a similar AppExpert application configuration.

To get started with an AppExpert application, you must first obtain the appropriate application template and import the template to the NetScaler appliance. After the AppExpert application is set up, you must verify that the application is working correctly. If required, you can customize the configuration to suit your requirements.

Periodically, you can verify and monitor the configuration by viewing the hit counters for various application components, statistics, and the Application Visualizer. You can also configure authentication, authorization, and auditing (AAA) policies for the application.

Updated: 2013-08-30

Following are the terms used in the AppExpert applications feature and the descriptions of the entities for which the terms are used:

Public Endpoint. The IP address and port combination at which the NetScaler appliance receives client requests for the associated web application. A public endpoint can be configured to receive either HTTP or secure HTTP (HTTPS) traffic. All client requests for the web application must be sent to a public endpoint. An AppExpert application can be assigned multiple endpoints. You configure public endpoints after you import a template.

Application Unit. An AppExpert application entity that processes a subset of web application traffic and load balances a set of services that host the associated content. The subset of traffic that an application unit must manage is defined by a rule. Each application unit also defines its own set of traffic optimization and security policies for the requests and responses that it manages. The NetScaler services associated with these policies are Compression, Caching, Rewrite, Responder, and application firewall.

By default, every AppExpert application with at least one application unit includes a default application unit, which cannot be deleted. The default application unit is not associated with a rule for identifying requests and is always placed last in the

order of application units. It defines a set of policies for processing any request that does not match the rules that are configured for the other application units, thereby ensuring that all client requests are processed.

Application units and their associated rules, policies, and actions are included in AppExpert application templates.

Service. The combination of the IP address of the server that hosts the web application instance and the port to which the application is mapped on the server, in the format <IP address>:<Port>. A web application that serves a large number of requests is usually hosted on multiple servers. Each server is said to host an instance of the web application, and each such instance of the web application is represented by a service on the NetScaler appliance. Services are deployment-specific, and are therefore not included in templates. You must configure services after you import a template.

Application Unit Rule. Either a classic expression or a default syntax expression that defines the characteristics of a traffic subset for an application unit. The following example rule is a default syntax expression that identifies a traffic subset that consists of four image types:

```
HTTP.REQ.URL.SUFFIX.EQ("bmp") || HTTP.REQ.URL.SUFFIX.EQ("gif") || HTTP.REQ.URL.SUFFIX.EQ("png") || HTTP.REQ.URL.SUFFIX.EQ("jpg")
```

For more information about default syntax expressions and classic policy expressions, see "Policies and Expressions."

Traffic Subset. A set of client requests that require a common set of traffic optimization and security policies. A traffic subset is managed by an application unit and is defined by a rule.

When the endpoint receives a client request, the NetScaler appliance evaluates the request against the rule that is configured for the topmost application unit. If the request satisfies this rule, the request is processed by the policies that are configured for the application unit, and then forwarded to a service. The choice of service depends on which services are configured for the application, and on settings such as the load balancing algorithm and persistence method configured for the application unit.

If the request does not satisfy the rule, the request is evaluated against the rule for the next topmost application unit. In this order, the request is evaluated against each application unit rule until the request satisfies a rule. If the request does not satisfy any of the configured rules, it is processed by the default application unit, which is always the last application unit.

You can configure multiple public endpoints for an AppExpert application. In such a configuration, by default, each application unit processes requests received by all the public endpoints and load balances all the services that are configured for the application. However, you can specify that an application unit processes traffic from only a subset of the public endpoints and load balances only a subset of the services that are configured for the AppExpert application.

Getting Started with an AppExpert Application

Aug 30, 2013

The process of setting up an AppExpert application begins with downloading the appropriate AppExpert application template from the Citrix Community Web site at "http://community.citrix.com/display/ns/AppExpert+Templates." The template that you need depends on the NetScaler release running on your appliance.

After you download the template, you must import the template to the NetScaler appliance, configure deployment settings, and then verify the configuration to make sure that the AppExpert application is working as expected.

Updated: 2013-08-30

You can either import the template file directly from your local computer or upload the template to the appliance and then import it. For more information about uploading a template to the NetScaler appliance, see "Uploading and Downloading Template Files."

During import, along with the template file that you specify in the AppExpert Template Wizard, you can include a deployment file that contains deployment details. If you choose to include a deployment file, you do not have to provide any additional information. All application-configuration information is imported from the template file, and all deployment-specific information for the application is imported from the deployment file. The NetScaler appliance imports all configuration settings from the deployment file through the NITRO API, and the wizard displays the configuration summary screen for your verification. If you do not include a deployment file, the wizard displays screens on which you can specify deployment information. During import, if an error occurs, any changes are automatically rolled back, preserving the configuration that was in place before you attempted to import the AppExpert application. For more information about the format of application templates and deployment files, see "Understanding NetScaler Application Templates and Deployment Files." For more information about how the template and deployment files are imported through the NITRO API, see "NITRO API."

AppExpert applications and the deployment files created from them support two or more endpoints. However, when importing an AppExpert template file, if you do not include a deployment file, the AppExpert Template Wizard displays a screen on which you can configure a maximum of two public endpoints—one endpoint of type HTTP and one endpoint of type HTTPS—for the application. So, if you want more than two endpoints, you have to configure additional endpoints after you create the application. You can then export the application to obtain a deployment file that contains all the configured endpoints.

If the public endpoint that you configure during import uses the HTTPS protocol, and you are not providing a deployment file, you must also specify a server certificate for the public endpoint. Additionally, if variables have been configured for the template, a Specify Variable Values page appears in the wizard. On this page, you can choose to specify new values for the variables. Configuring deployment settings (a minimum of one public endpoint and one or more services) during template import is not mandatory; you can choose to skip these steps during import and, instead, configure these settings after you import the template. Note, however, that you can configure additional AppExpert application features, such as AAA, only after you specify at least one public endpoint.

For more information about configuring endpoints after you import a template, see "Configuring Public Endpoints." For more information about configuring services and service groups after you import a template, see "Configuring Services and Service Groups." For more information about configuring variables for a NetScaler application, see "Creating Variables in

Application Templates."

To import an AppExpert application template to the NetScaler appliance

1. Navigate to AppExpert > Applications.
2. In the details pane, click Applications, and then click Import.
3. Follow the instructions in the AppExpert Template Wizard.

Updated: 2013-08-30

Verification is an important step in the process of setting up the NetScaler application. Before you proceed with other configuration tasks, you must verify that the state of the entities, such as endpoints and application units, are UP, and then test the entities for correct processing.

Verifying the Configuration

The graphical user interface (GUI) includes icons that indicate the states of the entities in the AppExpert application. These icons are displayed for applications and application units and are based on the health checks that the NetScaler appliance performs periodically on services and entities. The following table lists the icons and describes their meanings.

Table 1. Descriptions of State Indicator Icons

Icon	Entity	Indicates that
	Application	At least one public endpoint is up. The application will accept client requests from the public endpoints that are up.
	Application unit	The application unit is up. The application unit is up when at least one service or service group is up.
	Application	The public endpoint is out of service (disabled). This indicator is displayed when only one public endpoint is configured for the AppExpert application.
	Application	All the endpoints that are configured for the application are out of service. This indicator is displayed only when multiple endpoints are configured for the application.
	Application unit	All the services configured for the application unit are down.

You must ensure that the icons for each application and its application units are green at all times. If the icon that is displayed for an application is not green, verify that you have configured the public endpoints correctly. If the icon that is displayed for an application unit is not green, verify that the services are configured correctly. However, note that a green indicator does not mean that the state of all associated entities is UP. It only means that the application has sufficient

resources (endpoints and services) to serve client requests. To verify that the state of all associated entities is UP, check the health of all the entities on the statistics page for the application. For more information about viewing the application statistics page, see "Viewing Application Statistics."

Testing the Configuration by Using Hit Counters

You can test the configuration by sending test HTTP requests for web application content through the NetScaler appliance, and then verifying that the requests are being processed by the right application units, by viewing the hit counters for the various AppExpert application entities. For example, to verify that the endpoint is receiving requests, you can view the hit counter for the AppExpert application. To verify that the configured application unit rules are being matched as expected, you can view the hit counters for the AppExpert application units.

Note: To view hit counters for policies and actions that are configured for AppExpert applications, you must go to the associated feature node. For example, to view the hit counter for a Rewrite policy that is configured for an AppExpert application, you must go to the Rewrite feature node in the NetScaler configuration utility.
For a test example, consider an AppExpert application that includes an application unit called "WebPages" for processing web page content, and an application unit called "Images" for processing images. In this example, the rule that is configured for the WebPages application unit includes an expression that checks whether an HTML file is being requested. The Images application unit includes an expression that checks whether an image file is being requested.

Consider an HTML file called sitehome.html, located at /var/www/html/myapplicationpages/, on a backend server with an IP address of 192.0.2.10. In addition to HTML content, the HTML file also references images stored on the server. An HTTP request for the HTML file, sent directly to the server, would be as follows:

http://192.0.2.10/myapplicationpages/sitehome.html

To send a test request for this file through the NetScaler appliance, in the URL, replace the IP address of the server with the IP address of the public endpoint that is configured for the AppExpert application. For example, if the IP address of the public endpoint is 192.0.2.11, your test URL would be as follows:

http://192.0.2.11/myapplicationpages/sitehome.html

After you send the request, you must view the hit counter for the application to verify that the public endpoint received the request, view the hit counter for the WebPages application unit to verify that the request for the HTML file matched the rule configured for the application unit, and view the hit counter for the Images application unit to verify that the requests for the images matched the rule configured for the application unit.

For the application, the Hits dialog box displays the total number of requests received by each configured public endpoint. For an application unit, the Hits dialog box displays the number of requests that the application unit processed from each of the public endpoints, and the total hit count.

1. Navigate to AppExpert > Applications.
2. In the details pane, click the application or application unit for which you want to view the hit counter.
3. Click Hits.

Customizing the Configuration

Mar 28, 2012

After you verify that the AppExpert application is working correctly, you can customize the configuration to suit your requirements.

You can configure public endpoints and services for the AppExpert application and specify only a subset of the endpoints and services for each application unit. When you want the AppExpert application to manage a traffic subset that is not included in the template, you can either add an application unit for the new traffic subset or modify an existing application unit rule. You can also specify the order of evaluation of the traffic subsets that the AppExpert application manages.

Finally, you can modify the policies that the template provided. If the AppExpert application template does not include policies for a particular NetScaler feature, such as Rewrite or application firewall, you can configure your own policies.

The order in which you perform these tasks depends on your requirement. However, before you configure a service for an application, you must configure the service for the parent application.

Configuring Public Endpoints

Aug 30, 2013

If you did not specify a public endpoint when importing an AppExpert application, you can specify public endpoints after you create the application. You can configure one public endpoint of type HTTP and one public endpoint of type HTTPS for your AppExpert application.

If endpoints are already configured for the application, you can dissociate endpoints from the AppExpert application and delete any endpoints that you no longer need. Note that when you dissociate a public endpoint from the AppExpert application, the endpoint is automatically unbound from the associated application unit, but it is not deleted from the system.

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application for which you want to configure public endpoints, and then click Configure Public Endpoints.
3. In the Choose Public Endpoints dialog box for the application, do one of the following:
   - If the endpoints you want are listed in the dialog box, click the corresponding check boxes.
   - If you want to specify all the public endpoints, click Activate All.
   - If you want to dissociate endpoints from the AppExpert application, clear the corresponding check boxes.
   - If you want to create a new public endpoint, click Add. Then, in the Create public endpoint dialog box, configure endpoint settings, and then click OK.

     In the Create public endpoint dialog box, you can specify only the name, IP address, port, and protocol for the endpoint. You can specify additional endpoint settings after you create the public endpoint. To specify additional endpoint settings, after you create the endpoint, in the Choose Public Endpoints dialog box, click the endpoint, and then click Open. Then, in the Configure Public Endpoint dialog box, provide additional settings, and then click OK.

     For more information about the parameters in the Create public endpoint and Configure Public Endpoint dialog boxes, see "Content Switching."

   - If you want to modify a public endpoint, click the endpoint, and then click Open. Then, in the Configure Public Endpoint dialog box, modify settings for the endpoint, and then click OK.

     For more information about the parameters in the Configure Public Endpoint dialog box, see "Content Switching."
4. Click Close.

Configuring Endpoints for an Application Unit

Aug 30, 2013

When you configure multiple public endpoints for an AppExpert application, by default, all endpoints are bound to each application unit, and each application unit processes the requests received by all the endpoints. However, you can specify that a given application unit manages the traffic that is received by only a subset of the endpoints that are configured for the AppExpert application.

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application unit for which you want to specify public endpoints, and then click Configure Public Endpoints.
3. In the Choose Public Endpoints dialog box for the application unit, do one of the following:
   - If you are specifying endpoints for the application unit for the first time, clear the check boxes that correspond to the endpoints that you do not want to be bound to the application unit.
   - If you want to specify endpoints that are listed in the dialog box but not currently bound to the application unit, click the corresponding check boxes.
4. Click OK.

Configuring Services and Service Groups

Aug 30, 2013

When you configure a service or service group, you either modify an existing service or service group, or add new services to the AppExpert application. You add services or service groups if you did not specify them when you imported the application template. You also add services and service groups when you increase the number of servers that host instances of the application. You can configure a service and service group for an application unit only after you configure the service or service group for the AppExpert application.

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application for which you want to configure services or service groups, and then click Configure Backend Services.
3. In the Configure Backend Services dialog box, do one of the following:
   - To configure services, click the Services tab.
   - To configure service groups, click the Service Groups tab.
4. On the Service or Service Groups tab, do one of the following:
   - If the services or service groups that you want are listed on the tab, click the corresponding check boxes.
   - If you want to specify all the services or service groups, click Activate All.
   - If you want to create a new service or service group, click Add. Then, in the Create Service dialog box or Create Service Group dialog box, configure settings for the service or service group, respectively, and then click Create.
   - If you want to modify a service, click the service, and then click Open. Then, in the Configure Service dialog box or Create Service Group dialog box, configure settings for the service or service group, respectively, and then click OK.

For information about the settings in the Create Service, Configure Service, and Create Service Group dialog boxes, see "Load Balancing."

Configuring Services, Service Groups, and Load Balancing Parameters for an Application Unit

Aug 30, 2013

When you configure services and service groups for an AppExpert application, by default, all the services and service groups are bound to each application unit. However, depending on how you have configured your web application, the application resources that are managed by an application unit might be hosted on only some of the servers that are configured as services for the AppExpert application. Or, a set of servers might host content that is meant for the requests received at one or more specific public endpoints. In such scenarios, if all the services and service groups that are configured for the AppExpert application are associated with the application unit, a request that is forwarded to a server that does not host the requested content might not be served or might be served incorrect content. Therefore, you must ensure that each application unit is configured to manage only those services that can serve the requested content.

When configuring services and service groups for an application unit, you might choose to specify load balancing settings such as the weights that services must be assigned and the desired load balancing, persistence, and spillover methods. For more information about these settings, see Load Balancing.

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application unit for which you want to configure a service or service group, and then click Configure Backend Services.
3. In the Configure Backend Services dialog box, do one of the following:
   - To configure services, click the Services tab.
   - To configure service groups, click the Service Groups tab.
4. In the Services or Service Groups tab, do one of the following:
   - Clear the check boxes that correspond to the services or service groups that you do not want configured for the application unit. Make sure that the check boxes that correspond to the services or service groups that you want configured for the application unit are selected. Then, in the Weight column, specify the weight that you want to assign to each configured service.
   - To specify all services or service groups, click Activate All.
5. On the Method and Persistence and Advanced tabs, specify the desired parameters.
6. Click OK.

Creating Application Units

Aug 30, 2013

You might need to add application units for traffic subsets that are either specific to your web application implementation or not defined in the template. When creating an application unit, you must configure a rule for the application unit.

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application for which you want to add an application unit, and then click Add.
3. Click Create.

Configuring Application Unit Rules

Aug 30, 2013

You might want to configure an application unit rule to include or exclude certain types of traffic. When you configure the rule, you can also define the syntax of the expression.

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Applications.
2. In the details pane, right-click the application unit for which you want to modify the rule, and then click Open.
3. In the Configure Application Unit dialog box, do the following:
   1. To specify the format of the new expression, do one of the following:
      - To specify that you want to configure a classic expression in the Rule box, click Classic Syntax.
      - To specify that you want to configure an advanced expression in the Rule box, click Default Syntax.
   2. In the Rule box, configure the expression.
4. Click OK.

Specifying the Order of Evaluation of Application Units

Aug 30, 2013

Application unit rules are evaluated in the order in which they are placed in the graphical user interface (GUI). The rule that is configured for the topmost application unit is always configured first, followed by the rule that is configured for the second topmost application unit, and so on. The default application unit is always evaluated last.

When a request matches the rule that is configured for an application unit, the request is processed by the application unit, and no further matching is performed. Therefore, the order of evaluation of application units becomes an important factor if the traffic subsets for two or more application units overlap. If the traffic subsets for two or more application units overlap, you must specify the order in which an incoming request is matched against the application unit rules.

1. Navigate to AppExpert > Applications.
2. In the details pane, do the following:
   - To move an application unit up by one step, right-click the application unit, and then click Move Up.
   - To move an application unit down by one step, right-click the application unit, and then click Move Down.

Configuring Policies for Application Units

Apr 24, 2014

For an AppExpert application, you can configure policies for Compression, Caching, Rewrite, Responder, and Application Firewall. The templates that you download from the Citrix Community web site provide you with a set of policies that fulfill the most common application management requirements. You might want to fine-tune or customize these policies. If the set of policies provided for a given application unit does not include policies for a particular feature, you can create and bind your own policies for that feature.

If you create an AppExpert application without using a template, you must configure all the policies that the web application needs.

The GUI uses various icons to indicate whether or not policies are configured for a feature. For an application unit, if a policy is configured for a given feature, an icon that represents the feature is displayed. For example, if a compression policy is configured for an application unit, a compression icon is displayed in the Compression column for the application unit. For features for which no policy is configured, an icon depicting a plus sign (+) is displayed.

Note: When configuring policies for application units, you might need to configure policies and expressions that are either in the classic or default syntax. Additionally, when you configure default syntax policies, you might need to specify parameters such as Goto expressions and invoke policy banks. For information about configuring policies and expressions in both formats, see "Policies and Expressions."

You can use either classic policies or advanced policies to configure compression, but you cannot bind compression policies of both types to the same application unit.

To configure a compression policy for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, in the row for the application unit you want to configure, click the icon provided in the Compression column.
3. In the Configure Compression Policies dialog box, do one or more of the following, depending on the configuration tasks you want to perform:
   - Click Switch to Default Syntax if you want to configure a default syntax compression policy. If you want to bind or configure classic compression policies, and if you are in the default syntax view, you can click Switch to Classic Syntax to return to the classic policy view and begin modifying bound classic policies or create and bind new classic compression policies.
     Important: This setting also determines what policies are displayed when you want to insert a policy. For example, if you are in the default syntax view, when you click Insert Policy, the list that appears in the Policy Name column will include only default syntax policies. You cannot bind policies of both types to an application unit.
   - If you want to configure classic policies, click either Request or Response, depending on whether you want the policy to be evaluated at request-time or at response-time.
     You can configure both request-time and response-time classic compression policies for an application unit. After evaluating all of the request-time policies, if no match is found, the appliance evaluates response-time policies.
   - To modify a compression policy that is already bound to the application unit, click the name of the policy, and then click Modify Policy. Then, in the Configure Compression Policy dialog box, modify the policy, and then click OK.

For information about modifying a compression policy, see "Compression."

- To unbind a policy, click the name of the policy, and then click Unbind Policy.
- To modify the priority assigned to a policy, double-click the priority value, and then enter a new value.
- To regenerate assigned priorities, click Regenerate Priorities.
- To insert a new policy, click Insert Policy and, in the list that is displayed in the Policy Name column, click New Policy. Then, in the Create Compression Policy dialog box, configure the policy, and then click Create.

  For information about modifying a compression policy, see "Compression."

- If you are configuring a default syntax expression, do the following:
  - In the Goto Expression column, select a Goto expression.
  - In the Invoke column, specify the policy bank that you want to invoke if the current policy evaluates to TRUE.

4. Click Apply Changes, and then click Close.

You can use only default syntax policies and expressions to configure Caching policies.

To configure Caching policies for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, in the row for the application unit you want to configure, click the icon provided in the Caching column.
3. In the Configure Cache Policies dialog box, do one or more of the following, depending on the configuration tasks you want to perform:
   - Click either Request or Response, depending on whether you want the policy to be evaluated at request-time or at response-time.

     You can configure both request-time and response-time Caching policies for an application unit. After evaluating all of the request-time policies, if no match is found, the appliance evaluates response-time policies.

   - To modify a Caching policy that is already bound to the application unit, click the name of the policy, and then click Modify Policy. Then, in the Configure Cache Policy dialog box, modify the policy, and then click OK.

     For information about modifying a Caching policy, see "Integrated Caching."

   - To unbind a policy, click the name of the policy, and then click Unbind Policy.
   - To modify the priority assigned to a policy, double-click the priority value, and then enter a new value.
   - To regenerate assigned priorities, click Regenerate Priorities.
   - To insert a new policy, click Insert Policy and, in the list that is displayed in the Policy Name column, click New Policy. Then, in the Create Cache Policy dialog box, configure the policy, and then click Create.

     For information about modifying a Caching policy, see "Integrated Caching."

   - In the Goto Expression column, select a Goto expression.
   - In the Invoke column, specify the policy bank that you want to invoke if the current policy evaluates to TRUE.

4. Click Apply Changes, and then click Close.

You can use only default syntax policies and expressions to configure Rewrite policies.

To configure Rewrite policies for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, in the row for the application unit you want to configure, click the icon provided in the Rewrite column.
3. In the Configure Rewrite Policies dialog box, do one or more of the following, depending on the configuration tasks you want to perform:
   - Click either Request or Response, depending on whether you want the policy to be evaluated at request-time or at response-time.

     You can configure both request-time and response-time Rewrite policies for an application unit. After evaluating all of the request-time policies, if no match is found, the appliance evaluates response-time policies.

   - To modify a Rewrite policy that is already bound to the application unit, click the name of the policy, and then click Modify Policy. Then, in the Configure Rewrite Policy dialog box, modify the policy, and then click OK.

     For information about modifying a Rewrite policy, see "Rewrite."

   - To unbind a policy, click the name of the policy, and then click Unbind Policy.
   - To modify the priority assigned to a policy, double-click the priority value, and then enter a new value.
   - To regenerate assigned priorities, click Regenerate Priorities.
   - To insert a new policy, click Insert Policy and, in the list that is displayed in the Policy Name column, click New Policy. Then, in the Create Rewrite Policy dialog box, configure the policy, and then click Create.

     For information about modifying a Rewrite policy, see "Rewrite."

   - In the Goto Expression column, select a Goto expression.
   - In the Invoke column, specify the policy bank that you want to invoke if the current policy evaluates to TRUE.
4. Click Apply Changes, and then click Close.

You can use only default syntax policies and expressions to configure Responder policies.

To configure Responder policies for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, in the row for the application unit you want to configure, click the icon provided in the Responder column.
3. In the Configure Responder Policies dialog box, do one or more of the following, depending on the configuration tasks you want to perform:
   - To modify a Filter policy that is already bound to the application unit, click the name of the policy, and then click Modify Policy. Then, in the Configure Responder Policy dialog box, modify the policy, and then click OK.

     For information about modifying a Responder policy, see "Responder."

   - To unbind a policy, click the name of the policy, and then click Unbind Policy.
   - To modify the priority assigned to a policy, double-click the priority value, and then enter a new value.
   - To regenerate assigned priorities, click Regenerate Priorities.

- To insert a new policy, click Insert Policy and, in the list that is displayed in the Policy Name column, click New Policy. Then, in the Create Responder Policy dialog box, configure the policy, and then click Create.

  For information about modifying a Responder policy, see "Responder."

- In the Goto Expression column, select a Goto expression.
- In the Invoke column, specify the policy bank that you want to invoke if the current policy evaluates to TRUE.

4. Click Apply Changes, and then click Close.

You can configure both classic and default syntax policies and expressions for Application Firewall. However, if a policy of one type is already bound globally or to a virtual server that is configured on the appliance, you cannot bind a policy of the other type to an application unit. For example, if a default syntax policy is already bound either globally or to a virtual server, you cannot bind a classic policy to an application unit.

To configure Application Firewall policies for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, in the row for the application unit you want to configure, click the icon provided in the Application Firewall column.
3. In the Configure Application Firewall Policies dialog box, do one or more of the following, depending on the configuration tasks you want to perform:
   - Click either Classic Expression or Advanced Expression depending on the type of expression you want to configure for the Application Firewall policy.
     Important: This setting also determines what policies are displayed when you want to insert a policy. For example, if you select Advanced Expression, when you click Insert Policy, the list that appears in the Policy Name column will include only default syntax policies. You cannot bind policies of both types to an application unit. This option is not available if a policy of either type is already bound either globally or to a virtual server.
   - To modify an application firewall policy that is already bound to the application unit, click the name of the policy, and then click Modify Policy. Then, in the Configure Application Firewall Policy dialog box, modify the policy, and then click OK.
     For information about modifying a application firewall policy, see "Policies."

   - To unbind a policy, click the name of the policy, and then click Unbind Policy.
   - To modify the priority assigned to a policy, double-click the priority value, and then enter a new value.
   - To regenerate assigned priorities, click Regenerate Priorities.
   - To insert a new policy, click Insert Policy and, in the list that is displayed in the Policy Name column, click New Policy. Then, in the Create Application Firewall Policy dialog box, configure the policy, and then click Create.
     For information about modifying a application firewall policy, see "Policies."

4. Click Apply Changes, and then click Close.

Configuring Persistency Groups for Application Units

Aug 30, 2013

You can configure a persistency group for the application units in an AppExpert application. In the context of an AppExpert application, a persistency group is a group of application units that you can treat as a single entity for the purpose of applying common persistence settings. When the application is exported to an application template file, the persistency group settings are included, and they are automatically applied to the application units when you import the AppExpert application.

1. Navigate to AppExpert > Applications.
2. In the Applications View dialog box, click the name of the application for whose application units you want to configure a persistency group, and then click Configure Persistency Groups.
3. In the Configure Persistency Groups dialog box, do one of the following:
   - To add a persistency group, click Add.
   - To modify a persistency group, click Open.
4. In the Create Persistency Group or Configure Persistency Group dialog box, set the following parameters:
   - Group Name*—Name of the persistency group. For the NetScaler appliance to recognize the persistency group as part of the application's configuration, the name of the AppExpert application must be included in the name of the persistency group, as a prefix. Therefore, by default, the appliance displays the prefix in the Group Name box, and you cannot remove that prefix. Enter a name of your choice after the prefix.
   - Persistence—Type of persistence for the virtual server. If you select SOURCEIP, in the IPv4 Netmask box, enter a network mask that specifies the number of bits that the appliance must consider when creating persistence sessions. If you select COOKIEINSERT, in the Cookie Domain and Cookie Name boxes, specify a domain attribute to send in the Set-Cookie directive, and a name for the cookie, respectively.
   - Timeout—Time period for which a persistence session is in effect.
   - Backup Persistence—Type of backup persistence for the group.
   - Backup Timeout—Time period, in minutes, for which backup persistence is in effect.
   - Application Units—To add an application unit to the persistency group, in the Available Application Units box, click the application unit, and then click Add. To remove an application unit from the persistency group, in the Configured Application Units box, click the application unit, and then click Remove.
5. Click OK.

Viewing AppExpert Applications and Configuring Entities by Using the Application Visualizer

Aug 30, 2013

The Application Visualizer is a graphical representation of an AppExpert application. The Visualizer displays the public endpoints, application units, backend services, and policies that are configured for the application. You can use the Visualizer to obtain a visual overview of an AppExpert application's configuration and configure some of the displayed entities. By default, the Visualizer displays application units, services, and monitors for the selected application.

1. Navigate to AppExpert > Applications.
2. In the details pane, click the name of the application that you want to view, and then click Visualizer.
3. Do one or more of the following:
   - To optimize the display area, choose Best Fit, Zoom In, or Zoom out. If an item that you want to see disappears from view after zooming in, you can click and drag the viewable area.
   - To save the graph as an image file, click Save Image.
   - To find a particular entity, in the Search in field, type an entity name. In the view area, the entity names that begin with the search string are highlighted. To restrict the search, click the drop-down menu and select the specific entity that you want to search for.
   - To view the policies that an application uses, click one or more icons to display feature-specific policies. The policy types are Compression, Filter, Rewrite, Responder, Cache, application firewall, Authorization, Auditing, HTML Injection, SureConnect, Priority Queuing, and Traffic.
   - To view the rule that is configured for an application unit, click the curve that connects the public endpoint to the application unit. The rule is displayed on the Related Tasks tab.
   - To view the binding information for an application unit, policy, or monitor, click the displayed icon, click the Related Tasks tab, and then click Show Bindings
   - To view member services, click the icon for the service, click the Related Tasks tab, and then click Show Member Services.
   - To view detailed statistics for a public endpoint or application unit, click the icon that is displayed, click the Related Tasks tab, and then click Statistics.
   - To view the Load Balancing Visualizer for an application unit, click the application unit, click Related Tasks, and then click Visualizer.
   - To view the number of requests received per second at a given point in time by the load balancing virtual server and the number of hits per second at a given point in time for rewrite, responder, and cache policies, click Show Stats. The statistical information is displayed on the respective nodes in the Visualizer. This information is not updated in real time and has to be refreshed manually. To refresh this information, click Refresh Stats.

1. Navigate to AppExpert > Applications.
2. In the details pane, click the name of the application that you want to configure or view, and then click Visualizer.
3. Do one or more of the following:
   - To configure an entity that is displayed in the viewing area, click the icon for the entity, click the Related Tasks tab, and then click Modify public endpoint.
     In the Application Visualizer, you can modify only public endpoints and services.

- To bind additional monitors to a service, click the Available Resources tab, select Monitors from the drop-down list, and then click and drag a monitor to a service.
- To unbind a service from an application unit, click the curve that connects the application unit and the service, click Related Tasks, and then click Unbind.
- To unbind a monitor from a service, click the curve that connects the service and the monitor, click Related Tasks, and then click Unbind.
- To modify a monitor, click the monitor, click Related Tasks, and then click Open.
- To modify the binding parameters for a monitor, click the curve that connects the monitor to the associated service, click Related Tasks, and then click Modify Parameters.
- To apply a common service configuration across multiple service containers that are displayed when the services bound to a vserver do not have the same configuration, click the service container whose configuration you want to apply to all the containers, and then, in Related Tasks, click Apply Configuration.
- To view a comparative list of the parameters whose values differ across service containers, click the icon for a container, click the Related Tasks tab, and then click Service Attributes Diff. The comparative list helps you determine which service container has the service configuration that you want to apply to all the containers. After you determine which service container has the configuration you want, right-click the container, and then click Apply this Configuration.
- To copy the configuration of an entity (other than the configuration of the AppExpert application) to the local computer's clipboard, click the entity, click the Related Tasks tab, and then click Copy Properties. You can then paste the configuration information in a word processing document or spreadsheet.

Monitoring a NetScaler Application

Mar 18, 2011

After you customize the AppExpert application, you can view application statistics to make sure that the application and all its entities are working correctly. You can also use the Application Visualizer to monitor statistics associated with certain entities such as policies and virtual servers.

You can also view the hit counters for various entities at regular intervals to make sure that counters are being updated.

Updated: 2013-08-30

In the Applications node, you can select an application and view the Statistics page for the application. On the Statistics page, you can monitor the health and states of public endpoints and application units, and view the following statistical information:

- Requests and responses per second for each of the public endpoints and application units.
- Bytes per second, at each endpoint, for incoming and outgoing traffic.
- Application unit hit counters and the number of client and server connections for each application unit.
- Statistics for the services that are bound to the application units.

On the Statistics page, you can also view CPU usage, memory usage, and system logs.

To view statistics for an application

1. Navigate to AppExpert > Applications.
2. In the details pane, click the application for which you want to view statistics, and then click Statistics.

Updated: 2013-08-30

You can use the Application Visualizer to monitor the number of requests received per second at a given point in time by the vservers and the number of hits per second at a given point in time for Rewrite, Responder, and Cache policies.

To view statistical information for vservers, Rewrite policies, Responder policies, and Cache policies in the Visualizer

1. Navigate to AppExpert > Applications.
2. In the details pane, select the application for which you want to view statistical information, and then click Visualizer.
3. In the Application Visualizer window, do the following:
   - To view the statistics, click Show Stats.

     The statistical information is displayed on the respective nodes in the Visualizer. This information is not updated in real time and has to be refreshed manually.

   - To refresh the statistical information, click Refresh Stats.

Updated: 2013-08-30

The hit counters that are provided for various AppExpert application entities enable you to monitor the functioning of public endpoints and application units. For an application, the Hits dialog box displays the total number of requests received by each configured public endpoint. For an application unit, the Hits dialog box displays the number of requests that the application unit processed from each of the public endpoints and the total hit count. For instructions on viewing hit counters, see "Verifying and Testing the Configuration."

Deleting an Application

Aug 30, 2013

If you no longer need an application and its application units, you can delete it. When you delete an AppExpert application, backend services are not deleted, and any public endpoints that the application used become available for use by other applications.

When deleting an application, you are also prompted to specify whether you want to delete any bound policies and actions that are not used elsewhere.

1. Navigate to AppExpert > Applications.
2. In the details pane, click the name of the application that you want to delete, and then click Remove.

Configuring Authentication, Authorization, and Auditing

May 21, 2015

You can configure Authentication, Authorization, and Auditing (AAA) for the applications that you configure on the appliance. An authentication policy that is configured for an application defines the type of authentication to apply when a user or group attempts to access the application. If external authentication is used, the policy also specifies the external authentication server. Authorization policies configured for an application specify whether a particular user or group can access the application. Auditing policies define the audit log type, the level at which logging is performed, and other audit server settings. Authentication and auditing policies use the classic policy format.

Authentication policies, authorization policies, and auditing policies can be configured in any order. However, before you configure AAA for an application, you must configure a public endpoint for the application.

This document includes the following deatils:

- Configuring Authentication
- Configuring Authorization
- Configuring Auditing
- Disabling AAA for an Application

Updated: 2013-08-30

Configuring authentication for an application involves specifying an authentication FQDN, an authentication virtual server, a server certificate, and authentication and session policies. Authentication policies are automatically bound to the authentication virtual server specified for the application.

To configure authentication for an AppExpert application

1. Navigate to AppExpert > Applications.
2. In the details pane, click the name of the application for which you want to configure authentication, and then click Authentication.
3. In the Authentication Wizard, on the Introduction page, click Next.
4. Follow the instructions in the Authentication Wizard.

Updated: 2013-08-30

You can configure authorization for users and groups to enable then to access an AppExpert application. If the AAA user or group for which you want to configure permissions has not already been created, you can create it from AppExpert and then configure permissions for application access.

To configure permissions for a AAA user or group to access an AppExpert application

1. Navigate to AppExpert > Applications.

2. In the details pane, click the AppExpert application for which you want to configure user or group access, and then click Authorization.
3. Do one of the following:
   - If the AAA user or group for which you want to configure permissions is already in the Groups/Users tree, drag the user or group from the Groups/Users tree to the Users or Groups node in the application tree. Then, right-click the user or group and click Allow.
   - If the AAA user or group for which you want to configure permissions is not configured on the appliance, in the application tree, right-click Users or Groups, and then click Add. In the Create AAA Group or Create AAA User dialog box, fill in the values, click Create, and then click Close.

     The user or group is created with the permission set to Allow. To change the permission setting, right-click the group or user, and then click the permission setting.
4. Click Close.

Updated: 2013-08-30

When you configure auditing policies for an application, you must specify the server to which the log messages must be directed, the format of the messages logged, and the log level. Optionally, you can configure other settings, such as the log facility and date format. Auditing policies are automatically bound to all the AppExpert application's public endpoints.

To configure auditing policies for an application

1. Navigate to AppExpert > Applications.
2. In the details pane, click the application for which you want to configure auditing policies, and then click Auditing.
3. In the Configure Auditing Policies dialog box, click Insert Policy.
   - To specify an existing auditing policy, under Policy Name, click the name of the policy, and then do the following:
     - To modify the priority that is assigned to the policy by default, under Priority, double-click the priority, and then type a new priority value.
     - To modify the settings of the audit server, under Server, double-click the name of the server, and then, in the Configure Auditing Server dialog box, modify the settings as appropriate. You can modify all the settings in this dialog box except the name of the audit server and the audit type. For more information about the settings in the Configure Auditing Server dialog box, see "Auditing Policies."
   - To create a new auditing policy, under Policy Name, click New Policy, and then, in the Create Auditing Policy dialog box, do the following:
     - In the Name box, type a name for the policy.
     - The Name box already contains the string that is required at the beginning of the server name. You cannot modify the string.
     - From the Auditing Type list, select the auditing type (either SYSLOG or NSLOG).
     - If the audit server you want to specify is already listed in the Server list, select the server from the list, and then, if you want to modify the server settings, click Modify. In the Configure Auditing Server dialog box, modify the settings as appropriate, and then click OK. For more information about the settings in the Configure Auditing Server dialog box, see "Auditing Policies."
     - If you want to configure a new audit server, click New, and then, in the Create Auditing Server dialog box, type a name for the server, specify the server IP address, port number, and other settings as appropriate. When finished,

click OK.
    - Click Create.
- To change the priorities for the new auditing policies you created, under Priority, for each policy for which you want to change the priority, double-click the priority value and type new priority value.
- To regenerate priorities, click Regenerate Priorities.
- To unbind a policy, click the policy, and then click Unbind Policy.
- To modify a policy, click the policy, and then click Modify Policy.

4. Click Apply Changes, and then click Close.

Updated: 2013-08-30

After you configure AAA for an application, you can disable the AAA configuration for that application. When you disable AAA for an application, the configuration is not lost. You can enable AAA for the application when you want to reapply the configuration.

To enable or disable AAA for an application

1. Navigate to AppExpert > Applications.
2. In the details pane, click the application for which you want to enable or disable AAA, and then do one of the following:
    - To disable AAA for the application, click Turn Off AAA.
    - To enable AAA for the application, click Turn On AAA.

Setting Up a Custom NetScaler Application

May 21, 2015

If an AppExpert application template is not available for the Web application that you want to manage through the NetScaler appliance, or if available AppExpert application templates do not suit your requirements, you can create an AppExpert application without a template.

To create an AppExpert application without a template, you must first create an application and application units. Then, you configure public endpoints, services, and service groups. Finally, you configure the policies that determine how application traffic is evaluated and processed.

After you create the application and application units and configure policies, you must verify the configuration and test it to make sure that it is working correctly, just as you would when you configure an application by using a prebuilt AppExpert application template. Then, you must monitor the application to make sure that the application and its entities are working correctly.

This document includes the following details:

- Creating an Application
- Creating Application Units
- Configuring Public Endpoints for an AppExpert Application
- Configuring Public Endpoints for an Application Unit
- Configuring Services and Service Groups for an AppExpert Application
- Configuring Services and Service Groups for an Application Unit
- Configuring Policies

Updated: 2013-08-30

When you create an AppExpert application, the appliance creates a container to which you can add application units. The default application unit is not created until you create the first application unit.

To create an AppExpert application

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click Applications, and then click Add.
3. In the Create Application dialog box, in Name, enter a name for the application, and then click OK.

Updated: 2013-08-30

For each subset of traffic associated with your web application, you must create an application unit.

To create an application unit for the AppExpert application

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application for which you want to add an application unit, and then click Add.
3. Click Create.

Updated: 2013-08-30

After you have created all the application units that you require, you must configure one or more public endpoints to enable clients to access the web application through the NetScaler appliance.

To configure public endpoints for an AppExpert application

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application for which you want to configure public endpoints, and then click Configure Public Endpoints.
3. In the Choose Public Endpoints dialog box for the application, do one of the following:
   - If the endpoints you want are listed in the dialog box, click the corresponding check boxes.
   - If you want to specify all the public endpoints, click Activate All.
   - If you want to dissociate endpoints from the AppExpert application, clear the corresponding check boxes.
   - If you want to create a new public endpoint, click Add. Then, in the Create public endpoint dialog box, configure endpoint settings, and then click OK.

     In the Create public endpoint dialog box, you can specify only the name, IP address, port, and protocol for the endpoint. You can specify additional endpoint settings after you create the public endpoint. To specify additional endpoint settings, after you create the endpoint, in the Choose Public Endpoints dialog box, click the endpoint, and then click Open. Then, in the Configure Public Endpoint dialog box, provide additional settings, and then click OK.

     For more information about the parameters in the Create public endpoint and Configure Public Endpoint dialog boxes, see "Content Switching."

   - If you want to modify a public endpoint, click the endpoint, and then click Open. Then, in the Configure Public Endpoint dialog box, modify settings for the endpoint, and then click OK.

     For more information about the parameters in the Configure Public Endpoint dialog box, see "Content Switching."

4. Click Close.

Updated: 2013-08-30

For an application unit, you specify public endpoints in the same way as you would specify public endpoints for an application that is created from an AppExpert application template. For more information about specifying a subset of the endpoints for an application unit, see "Configuring Endpoints for an Application Unit."

To configure endpoints for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application unit for which you want to specify public endpoints, and then click Configure Public Endpoints.
3. In the Choose Public Endpoints dialog box for the application unit, do one of the following:
   - If you are specifying endpoints for the application unit for the first time, clear the check boxes that correspond to the endpoints that you do not want to be bound to the application unit.
   - If you want to specify endpoints that are listed in the dialog box but not currently bound to the application unit, click

the corresponding check boxes.
4. Click OK.

Updated: 2013-08-30

Services and service groups are available for application units only after you configure the services and service groups for the AppExpert application. Therefore, you must configure services and service groups for the AppExpert application before you configure the services for the application units. All the services and service groups that you configure for an AppExpert application must use the same protocol (either HTTP or HTTPS). The procedure for configuring services and service groups for an AppExpert application that is not created from a template is the same as that for an application created from a template.

To configure a service or service group for the AppExpert application

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application for which you want to configure services or service groups, and then click Configure Backend Services.
3. In the Configure Backend Services dialog box, do one of the following:
   - To configure services, click the Services tab.
   - To configure service groups, click the Service Groups tab.
4. On the Service or Service Groups tab, do one of the following:
   - If the services or service groups that you want are listed on the tab, click the corresponding check boxes.
   - If you want to specify all the services or service groups, click Activate All.
   - If you want to create a new service or service group, click Add. Then, in the Create Service dialog box or Create Service Group dialog box, configure settings for the service or service group, respectively, and then click Create.
   - If you want to modify a service, click the service, and then click Open. Then, in the Configure Service dialog box or Create Service Group dialog box, configure settings for the service or service group, respectively, and then click OK.

For information about the settings in the Create Service, Configure Service, and Create Service Group dialog boxes, see "Load Balancing."

Updated: 2013-08-30

After you configure services and service groups, you must configure services and service groups for each application unit. However, this step is not necessary if each backend service hosts all the content associated with the web application. You configure services and service groups for an application unit if the content associated with the application unit is hosted on only a subset of the backend servers.

To configure services or service groups for an application unit

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application unit for which you want to configure a service or service group, and then click Configure Backend Services.
3. In the Configure Backend Services dialog box, do one of the following:
   - To configure services, click the Services tab.
   - To configure service groups, click the Service Groups tab.

4. In the Services or Service Groups tab, do one of the following:
   - Clear the check boxes that correspond to the services or service groups that you do not want configured for the application unit. Make sure that the check boxes that correspond to the services or service groups that you want configured for the application unit are selected. Then, in the Weight column, specify the weight that you want to assign to each configured service.
   - To specify all services or service groups, click Activate All.
5. On the Method and Persistence and Advanced tabs, specify the desired parameters.
6. Click OK.

Updated: 2013-08-30

The procedures for configuring policies for an AppExpert application that is created without using a template are the same as those for an AppExpert application that was created from a template. For more information, see "Configuring Policies for Application Units."

Creating and Managing Template Files

May 10, 2012

After you set up an AppExpert application and customize it to suit your requirements, you can create a template from the application and then share the template with other administrators. Or, you can create a template and then import the template to other NetScaler appliances that require a similar AppExpert application configuration. This simplifies and expedites the process of setting up similar applications on other appliances. You can also export a content switching configuration to a template file. When creating a template file, you can configure variables in the policy expressions and actions that are configured for an application.

AppExpert application template files can be exported either to the template directory on the NetScaler appliance or to a folder on your local computer. You can then upload and download the templates to and from the NetScaler appliance and rename the templates that are stored in the AppExpert application templates directory on your appliance.

This document includes the following information:

- Exporting an AppExpert Application to a Template File
- Exporting a Content Switching Virtual Server Configuration to a Template File
- Creating Variables in Application Templates
- Uploading and Downloading Template Files
- Understanding NetScaler Application Templates and Deployment Files

Exporting an AppExpert Application to a Template File

Sep 30, 2013

When you export an AppExpert application, all application-configuration information is exported to a template file and all deployment-specific information is exported to a deployment file. The string _deployment is automatically appended to the name of the template file to create the name of the deployment file. Both files are in XML format. If you choose to export the application template file to the NetScaler appliance, the template file is stored in the /nsconfig/nstemplates/applications directory on the NetScaler appliance and the deployment file is stored in the /nsconfig/nstemplates/applications/deployment_files/ directory. For more information about the format of application templates and deployment files, see "Understanding NetScaler Application Templates and Deployment Files". If you have configured NetScaler Gateway application, when you export the AppExpert configuration, you can choose to include the NetScaler Gateway policies in the template.

1. Navigate to AppExpert > Applications.
2. In the details pane, click the name of the application that you want to export as a template file, and then click Export.
3. In the Export...as Template dialog box, do the following:
   1. In the Name box, modify the name of the template, if necessary.
   2. If you want to configure variables for the template, click Configure Variables, and then, in the Configure Variables dialog box, configure the variables that you want.
      For more information about configuring variables in application templates, see "Creating Variables in Application Templates".
   3. If you want to export the template file to the application templates directory on the appliance, make sure that Browse (Appliance) is displayed.
   4. If you want to export the template file to your computer, click the Browse (Appliance) drop-down menu, click Local, browse to the location to which you want to save the file, and then click Save.
   5. Provide the following information:
      - **Introduction Description**—Any text that introduces the AppExpert application template during import. This text is displayed on the Specify Application Name page of the AppExpert Template Wizard when the template is imported.
      - **Summary Description**—Any summary that you might want to display on the Summary page of the AppExpert Template Wizard when the template is imported.
      - **Author**—The name of the author of the template.
      - **Major**—The major version number of the template.
      - **Minor**—The minor version number of the template. This number is appended to the major version number and displayed on the Summary page of the AppExpert Template Wizard, during import, in the format Major.Minor.
   6. Click OK.
      If NetScaler Gateway policies have been configured for the application, you will be prompted to include the NetScaler Gateway configuration in the application template. If you want to include the NetScaler Gateway configuration in the template, at the prompt, click Yes.

Exporting a Content Switching Virtual Server Configuration to a Template File

Aug 30, 2013

You can also export a content switching configuration as an application template. You can export a content switching virtual server configuration to an application template either from the Content Switching Virtual Servers pane or from the Content Switching Visualizer. Configuration information, which includes the content switching virtual server, all associated load balancing virtual servers, services, service groups, and policies, is exported to a template file and all deployment-specific information is exported to a deployment file. The string "_deployment" is automatically appended to the name of the template file to create the name of the deployment file. Both files are in XML format. If you choose to export the application template file to the NetScaler appliance, the template file is stored in the /nsconfig/nstemplates/applications directory on the NetScaler appliance and the deployment file is stored in the /nsconfig/nstemplates/applications/deployment_files/ directory. For more information about the format of application templates and deployment files, see "Understanding NetScaler Application Templates and Deployment Files." The configuration information that is exported includes the content switching virtual server, all associated load balancing virtual servers, services, service groups, and policies.

However, if the content switching virtual server is already configured as the public endpoint for an AppExpert application, you cannot export the configuration to a template file . In this scenario, you must export the associated AppExpert application to a template. For more information about exporting an AppExpert application to a template file, see "Exporting an AppExpert Application to a Template File."

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, click the name of the content switching virtual server whose configuration you want to export as a template file, and then click Visualizer.
3. In the Content Switching Visualizer, click the icon for the content switching vserver, click Related Tasks, and then click Create Template.
4. In the Export...as Template dialog box, enter a name for the template file, and then do one of the following:
   - To export the template file to the appliance, make sure that Browse (Appliance) is displayed.
   - To export the template file to your computer, click the Browse (Appliance) drop-down menu, click Local, browse to the location to which you want to save the file, and then click Save.
5. Provide the following information:
   - **Introduction Description**—Any text that introduces the AppExpert application template during import. This text is displayed on the Specify Application Name page of the AppExpert Template Wizard when the template is imported.
   - **Summary Description**—Any summary that you might want to display on the Summary page of the AppExpert Template Wizard when the template is imported.
   - **Author**—The name of the author of the template.
   - **Major**—The major version number of the template.
   - **Minor**—The minor version number of the template. This number is appended to the major version number and displayed on the Summary page of the AppExpert Template Wizard, during import, in the format `Major.Minor`.
6. Click OK.

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, click the name of the content switching virtual server whose configuration you want to export as a template file, and then click Create AppExpert Template.
3. Perform steps 4 through 6 described in "To export a content switching configuration to an application template file from the Content Switching Visualizer".

Creating Variables in Application Templates

Aug 30, 2013

Application templates support the declaration of variables in the policy expressions and actions that are configured for an application. The ability to declare variables in policy expressions and actions enables you to replace preconfigured values in expressions (for example, configurable parameters such as the host name of a server or the target for a Rewrite action) with values that suit the environment into which you are importing the template. If variables have been configured for an AppExpert application template, the AppExpert Template Wizard, which appears when you import an AppExpert application template, includes a Specify Variable Values page on which you can specify appropriate values for the variables that are configured for the template.

As an example, consider the following policy expression that is configured to evaluate the value of the Host header in an HTTP request:

```
HTTP.REQ.HEADER("Host").CONTAINS("server1")
```

If you want the server name to be configurable at import time, you can specify the string "server1" as a variable. When importing the template, you can specify a new value for the variable on the Variables tab.

After you create a variable, you can do the following:

- Assign additional strings to an existing variable. After you create a variable for a string, you can select and assign other parts of the same or different expression to the variable. The strings you assign to a variable need not be the same. At import time, all the strings that are assigned to the variable are replaced with the value that you provide.
- View the string or strings that are assigned to the variable.
- View a list of all the entities and parameters that use the variable.

In the export application template wizard, you can define variables in certain fields (fields with an adjacent  button) for the following entities:

- Cache policies
- Rewrite policies
- Rewrite actions
- Responder policies
- Responder actions

To configure a variable in a policy expression or action

1. Navigate to AppExpert > Applications.
2. In the details pane, right-click the application that you want to export to a template file, and then click Export.
3. In the Export...as Template dialog box, modify the default template file name if required, specify the location where you want to save the template, and then click Configure Variables.
4. In the Configure Variables dialog box, click the tab that lists the policy expression or action for which you want to configure a variable, select the expression, and then click Configure Variables.
5. In the Variables dialog box, click the  button next to the expression or value in which you want to create a variable.
6. In the Variables dialog box, do the following:
   - To create a variable, in the text box that displays the configured expression or value, select the string that you want to be configurable at import time, and then click Add. In the Add Variable dialog box, specify a name and a description for the variable, and then click Create.

- The name of the variable, its value, and the description you provided appear in the Available Variables listing in the dialog box. The name you provide will be the name of the associated field in the template import wizard, and the description will appear as alt text when the user positions the mouse pointer over the field.
- To modify a variable, in the Available Variables list, click the variable, and then click Open. In the Add Variable dialog box, modify the value and the description, and then click OK.
- To view all the strings that are assigned to a given variable, in the Available Variables listing, click the name of the variable. The strings that are assigned to the variable are highlighted.
- To view a list of all the entities and parameters in which the variable is used, in the Available Variables listing, click the variable whose references you want to view, and then click Show References.
- To assign a string to an existing variable, in the text box that displays the expression you configured, select the string you want to assign to an existing variable, right-click the selection, click Use Existing Selection, and then click the name of the variable to which you want to assign the string.

  If a variable has multiple strings assigned to it, when you specify a new value for the variable during import, all strings assigned to the variable are replaced with the new value.

7. Click Close.

Uploading and Downloading Template Files

Aug 30, 2013

Template files can be uploaded from your local computer to the NetScaler appliance or downloaded from the appliance to your local computer. On the appliance, AppExpert application templates are always stored in the AppExpert application templates directory, which is /nsconfig/nstemplates/applications/.

To upload an AppExpert application template from your local computer to the NetScaler appliance

1. Navigate to AppExpert > Templates.
2. In the details pane, click Manage Templates.
3. In the Manage Application Templates dialog box, click Application Templates, and then click Upload.
4. In the Upload Application Template dialog box, browse to the directory in which the template file is stored, click the template file, and then click Select.
   The template file is uploaded to the AppExpert application template directory on the appliance.

To download an AppExpert application template from the NetScaler appliance to your local computer

1. Navigate to AppExpert > Templates.
2. In the details pane, click Manage Templates.
3. In the Manage Application Templates dialog box, click the AppExpert application template that you want to download, and click Download.
4. In the Download Application Template dialog box, browse to the location to which you want to save the file, and then click Save.

Understanding NetScaler Application Templates and Deployment Files

Mar 28, 2012

When you export a NetScaler application, the following two files are automatically created:

- **NetScaler application template file.** Contains application-configuration information such as application units, rules, and configured policies.
- **Deployment file.** Contains deployment-specific information such as public endpoints, services, associated IP addresses, and configured variables.

In the application template and deployment file, each unit of application-configuration information is encapsulated in a specific XML element that is meant for that unit type. For example, each public endpoint and associated endpoint details are encapsulated within the <appendpoint> and </appendpoint> tags, and all the endpoint elements are encapsulated within the <appendpoint_list> and </appendpoint_list> tags.
Note: After you export a NetScaler application, you can add elements, remove elements, and modify existing elements before importing the application to a NetScaler appliance.

Example of a NetScaler Application Template

Following is an example of a template file that was created from a NetScaler application called "SharePoint_Team_Site":

```
<?xml version="1.0" encoding="UTF-8" ?>
<template>
<template_info>
  <application_name>SharePoint_Team_Site</application_name>
  <templateversion_major>1</templateversion_major>
  <templateversion_minor>1</templateversion_minor>
  <author>Ed</author>
  <introduction>An application for managing a SharePoint team site with images, reports, and, XML content.</introduction>
  <summary>This template includes variables</summary>
  <version_major>9</version_major>
  <version_minor>3</version_minor>
  <build_number>38</build_number>
</template_info>
<apptemplate>
  <rewrite>
    <rewriteaction_list>
      <rewriteaction>
        <name>Rw_name</name>
        <type>replace</type>
        <target>HTTP.REQ.BODY(10000).AFTER_REGEX(re/number/).BEFORE_REGEX(re/address/)</target>
        <stringbuilderexpr>"NA"</stringbuilderexpr>
        <allow_unsafe_pi1>NO</allow_unsafe_pi1>
      </rewriteaction>
      <rewriteaction>
        .
        .
        .
      </rewriteaction>
      .
      .
      .
    </rewriteaction_list>
    <rewritepolicy_list>
      <rewritepolicy>
        <name>Rw_number_NA</name>
        <rule>HTTP.REQ.BODY(100000).CONTAINS("admin")</rule>
        <action>Rw_name</action>
      </rewritepolicy>
      <rewritepolicy>
```

```
      .
      .
      .
    </rewritepolicy>
    .
    .
    .
  </rewritepolicy_list>
</rewrite>
<appunit_list>
  <appunit>
    <name>SharePoint_Team_Sitedefault</name>
    <rule />
    <expressiontype>PE</expressiontype>
    <servicetype>HTTP</servicetype>
    <ipv46>0.0.0.0</ipv46>
    <ipmask>*</ipmask>
    <port>0</port>
    <range>1</range>
    <persistencetype>NONE</persistencetype>
    <timeout>2</timeout>
    <persistencebackup>NONE</persistencebackup>
    <backuppersistencetimeout>2</backuppersistencetimeout>
    <lbmethod>LEASTCONNECTION</lbmethod>
    <persistmask>255.255.255.255</persistmask>
    <v6persistmasklen>128</v6persistmasklen>
    <pq>OFF</pq>
    <sc>OFF</sc>
    <m>IP</m>
    <datalength>0</datalength>
    <dataoffset>0</dataoffset>
    <sessionless>DISABLED</sessionless>
    <state>ENABLED</state>
    <connfailover>DISABLED</connfailover>
    <clttimeout>180</clttimeout>
    <somethod>NONE</somethod>
    <sopersistence>DISABLED</sopersistence>
    <redirectportrewrite>DISABLED</redirectportrewrite>
    <downstateflush>DISABLED</downstateflush>
    <gt2gb>DISABLED</gt2gb>
    <ipmapping>0.0.0.0</ipmapping>
    <disableprimaryondown>DISABLED</disableprimaryondown>
    <insertvserveripport>OFF</insertvserveripport>
    <authentication>OFF</authentication>
    <authn401>OFF</authn401>
    <push>DISABLED</push>
    <pushlabel>none</pushlabel>
    <l2conn>OFF</l2conn>
  </appunit>
  <appunit>
    .
    .
    .
  </appunit>
  .
  .
  .
</appunit_list>
```

```
  </apptemplate>
  <parameters>
    <property_list>
      <property>
        <variable_definition_list>
          <variable_definition>
            <name>body_size</name>
            <defaultvalue>10000</defaultvalue>
            <description>Evaluation Scope</description>
            <startindex>14</startindex>
            <length>5</length>
          </variable_definition>
          .
          .
          .
        </variable_definition_list>
        <object_type>rewriteaction</object_type>
        <object_name>Rw_name</object_name>
        <name>target</name>
      </property>
      .
      .
      .
    </property_list>
  </parameters>
</template>
```

Example of a Deployment File

Following is the deployment file associated with the "SharePoint_Team_Site" application in the preceding example:

```
<?xml version="1.0" encoding="UTF8" ?>
<template_deployment>
  <template_info>
    <application_name>SharePoint_Team_Site</application_name>
    <templateversion_major>1</templateversion_major>
    <templateversion_minor>1</templateversion_minor>
    <author>Ed</author>
    <introduction>An application for managing a SharePoint team site with images, reports, and, XML content.</introduction>
    <summary>This template includes variables</summary>
    <version_major>9</version_major>
    <version_minor>3</version_minor>
    <build_number>38</build_number>
  </template_info>
  <appendpoint_list>
    <appendpoint>
      <ipv46>10.111.111.1</ipv46>
      <port>80</port>
      <servicetype>HTTP</servicetype>
    </appendpoint>
  </appendpoint_list>
  <service_list>
    <service>
      <ip>10.102.29.5</ip>
      <port>80</port>
      <servicetype>HTTP</servicetype>
    </service>
    <service>
      .
      .
      .
```

```
    </service>
    .
    .
    .
  </service_list>
  <variable_list>
    <variable>
      <name>body_size</name>
      <description>Evaluation Scope</description>
      <value>10000</value>
    </variable>
    <variable>
      .
      .
      .
    </variable>
    .
    .
    .
  </variable_list>
</template_deployment>
```

NetScaler Gateway Applications

May 21, 2015

When you configure an AppExpert application to manage a web application through the Citrix® NetScaler® appliance, you also create a set of application units and configure a set of traffic optimization and security policies for each unit. The policies that you configure for each application unit (policies for features such as Compression, Caching, and Rewrite) evaluate traffic that is meant only for that unit. In addition to these policies, you might want to configure Access Gateway policies for the application as a whole to optimize the application traffic when accessed through the Access Gateway. The Access Gateway Applications feature enables you to configure Access Gateway policies (Authorization, Traffic, Clientless Access, and TCP Compression) for an AppExpert application. After you configure NetScaler Gateway policies for AppExpert applications, you can include the policy configuration in the AppExpert application templates that you create.

You can also configure NetScaler Gateway policies for intranet subnets, file shares, and other network resources.

Finally, you can create bookmarks for AppExpert applications and certain resources if you want users to be able to access them from the NetScaler Gateway home page.

You can configure the entities in the NetScaler Gateway Applications feature only by using the configuration utility.

How an NetScaler Gateway Application Works

Updated: 2013-07-17

When you create an AppExpert application in the Applications node in the configuration utility, a corresponding Access Gateway application is automatically created in the Access Gateway Applications node. Additionally, a rule that uses the AppExpert application's configured public endpoint is automatically created for the Access Gateway application entry. If multiple endpoints are configured for the AppExpert application, the rule includes all the configured public endpoints. The NetScaler appliance uses this rule to apply any configured Access Gateway policies to the traffic received at the AppExpert application's public endpoint. Traffic received at the AppExpert application's public endpoint is first evaluated against the NetScaler Gateway policies and then evaluated against the policies configured for AppExpert application's application units.

The rule that is created for the Clientless Access policies for an Access Gateway application is an advanced expression that also uses the public endpoint that is configured for the AppExpert application. Therefore, before you configure NetScaler Gateway policies for an AppExpert application, you must configure public endpoints for the AppExpert application.

When you include the NetScaler Gateway configuration in an application template, deployment-specific information, such as IP address and port information, and the rule that is created from this information are not included in the template.

How a NetScaler Configuration for a File Share Works

On the NetScaler appliance, you can configure Authorization policies for a file share that is hosted on your organization's network.

When you create a file share, you specify a name for the file share and the network path to the file share. In the network path, you can specify either the name of the server or the server IP address. A rule that uses the components of the file share path is automatically created for the file share. This rule enables the appliance to identify requests for files hosted on the file share server. Any Authorization policies that are configured for the file share are applied to incoming requests.

The NetScaler configuration for a file share cannot be saved in AppExpert application templates.

How a NetScaler Configuration for an Intranet Subnet Works

For the intranet subnets that form a part of your network, you can configure policies for Authorization, Traffic, and TCP Compression on the NetScaler appliance. When adding an intranet subnet, you specify the IP address and the netmask of the intranet subnet. A rule that uses these two parameters is automatically created for the intranet subnet. The appliance applies the configured policies to any request that has a destination IP address and netmask set to the subnet's IP address and netmask, respectively.

The NetScaler configuration for an intranet subnet cannot be saved in AppExpert application templates.

How the Other Resources Category Works

Updated: 2013-07-18

The Other Resources category enables you to configure Access Gateway policies for any network resource by using a rule of your choice. When you configure the NetScaler appliance to process requests for the network resource, you configure a classic expression to identify the requests that are associated with the network resource. You can configure Authorization, Traffic, Clientless Access, and TCP Compression policies for a network resource in Other Resources. The NetScaler appliance applies the configured NetScaler Gateway policies to any requests that match the configured rule.

The NetScaler configuration for a network resource in Other Resources cannot be saved in AppExpert application templates.

Entity Naming Conventions

The NetScaler Gateway Applications feature enforces a naming convention for some of the entities that you create in this feature. For example, the names of the profiles that you create for Traffic policies for an intranet subnet always begin with a string that consists of the name of the intranet subnet followed by an underscore (_). The name that you provide for the entity is appended to this string. If the name of a subnet is "subnet1," the name of the profile begins with "subnet1_." When such a naming convention is required (in the text box in which you type the name of an entity, for example), the user interface automatically inserts the string with which the name of the entity must begin and does not allow you to modify it.

Adding File Shares

Mar 28, 2012

When creating a file share, you provide the network path to the file share. Any policies that you configure for a file share use the rule that is automatically created when you created the file share.

To configure a file share

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, click File Shares, and then do one of the following:
   - To add a file share, click File Shares, and then click Add.
   - To modify a file share, click File Shares, and then click Open.
3. In the Create File Share or Configure File Share dialog box, do the following:
   1. In the Name box, type a name for the file share you are adding. This parameter cannot be changed for an existing file share.
   2. In the Path box, type the path to the file share.
      The path to the file share may use either the name of the server or the IP address of the server.
   3. In Bookmark, in the Text to Display box, type a name for the file share as you would want it to appear on the Access Gateway home page.
   4. Click Create or OK, and then click Close.

Adding Intranet Subnets

Mar 28, 2012

You can specify authorization and Traffic policies for traffic that is bound for the intranet subnets that are configured in your network. The rules for these policies are automatically created by using the parameters you specify for the subnet.

To configure an intranet subnet

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, do one of the following:
   - To add an intranet subnet, click Intranet Subnets, and then click Add.
   - To modify an intranet subnet, click an intranet subnet, and then click Open.
3. In the Create Intranet Subnet or Configure Intranet Subnet dialog box, do the following:
   1. In the Name box, type a name for the intranet subnet you are adding. This parameter cannot be changed for an existing intranet subnet.
   2. In the IP Address box, type the IP address of the intranet subnet.
   3. In the Netmask box, type the netmask that will be used for the intranet subnet.
   4. Click Create or OK, and then click Close.

Adding Other Resources

Jul 17, 2013

For a network resource that you add to Other Resources, you must configure a classic expression that identifies the subset of traffic associated with the resource. For more information about configuring a classic expression, see the Policy Configuration and Reference.

To configure a resource in Other Resources

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, do one of the following:
   - To add a resource, click Other Resources, and then click Add.
   - To modify a resource, click a resource, and then click Open.
3. In the Create Resource or Configure Resource dialog box, do the following:
   1. In the Name box, type a name for the resource you are adding. This parameter cannot be changed for an existing resource.
   2. In the Rule box, type the rule that will identify the subset of traffic that is associated with the resource you are adding.
      Alternatively, click Configure, and then create the rule in the Create Expression dialog box.
   3. Click Create or OK, and then click Close.

Configuring Authorization Policies

Mar 28, 2012

You can configure NetScaler Gateway authorization policies for AAA users and groups to access a resource.

To configure permissions for a AAA user or group to access a resource

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, in the Authorization column, click the icon for the application, file share, intranet subnet, or resource for which you want to configure authorization policies for AAA users and groups.
3. Do one of the following:
   - If the AAA user or group for which you want to configure permissions is already in the Groups/Users tree, drag the user or group from the Groups/Users tree to the Users or Groups node in the <application name> tree. Then, right-click the user or group and click Allow.
   - If the AAA user or group for which you want to configure permissions is not configured on the appliance, in the <application name> tree, right-click Users or Groups, and then click Add. In the Create AAA Group or Create AAA User dialog box, fill in the values, click Create, and then click Close.

     The user or group is created with the permission set to Allow. To change the permission setting, right-click the group or user, and then click the permission setting.
4. Click Close.

Configuring Traffic Policies

Dec 16, 2013

The traffic policies that you configure for the resources in the NetScaler Gateway Applications node control client connections to the application. You do not have to configure a rule for the resource. The rule created automatically when you create the resource. You only need to associate a request profile with the traffic policy. In the traffic profile, you specify parameters such as the protocol, application time-out, and file type association.

To configure traffic policies for a resource

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, in the Traffic column, click the icon provided for the application, file share, intranet subnet, or resource for which you want to configure traffic policies.
3. In the Configure Traffic Policies dialog box, do the following:
   - To specify an existing traffic policy, click Insert Policy, and then, in the Policy Name column, click the name of the policy.
   - To configure a new policy, click Insert Policy, and then, in the Policy Name column, click New Policy. In the Create Traffic Policy dialog box, in the Name box, after the underscore (_), type a name for the policy. Then, in Request Profile, either select an existing request profile or click New to configure a new request profile. You can also select an existing profile and then click Modify to modify the profile.

     For more information about configuring a traffic policy or profile, see NetScaler Gateway , Enterprise Edition at http://edocs.citrix.com/.

   - To modify a policy that you have inserted, in the Policy Name column, click the policy name, and then click Modify Policy. To modify only the associated profile, in the Profile column, click the name of the profile, and then click Modify Profile.
   - To regenerate the priorities assigned to the policies, click Regenerate Priorities.
   - To specify a new priority value for a policy, in the Priority column, double-click the assigned priority, and then enter the value you want.
   - To unbind a policy, click the policy, and then click Unbind Policy.
4. Click Apply Changes, and then click Close.

Configuring Clientless Access Policies

Dec 16, 2013

Clientless access, when configured for a resource on the NetScaler appliance, allows end-users to access the resource without using the NetScaler Gateway client software. Users can use web browsers to access resources such as Outlook Web Access. You configure clientless access for a resource by configuring a clientless access policy that is associated with a clientless access profile.

To configure a clientless access policy for a resource in the NetScaler Gateway Applications node

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, in the Clientless Access column, click the icon for the application, file share, intranet subnet, or resource for which you want to configure a clientless access policy.
3. In the Configure Clientless Access Policies dialog box, do the following:
   - To specify an existing clientless access policy, click Insert Policy, and then, in the Policy Name column, click the name of the policy.
   - To configure a new clientless access policy, click Insert Policy, and then, in the Policy Name column, click New Policy. In the Create Clientless Access Policy dialog box, in the Name box, after the underscore (_), type a name for the policy. Then, in Profile, either select an existing profile or click New to configure a new profile. You can also select an existing profile and then click Modify to modify the profile.

     For more information about configuring a clientless access policy or profile, see NetScaler Gateway , Enterprise Edition at http://edocs.citrix.com/.

   - To modify a policy that you have inserted, in the Policy Name column, click the policy name, and then click Modify Policy. To modify only the associated profile, in the Profile column, click the name of the profile, and then click Modify Profile.
   - To specify a new priority value for a policy, in the Priority column, double-click the assigned priority, and then enter the value you want.
   - To unbind a policy, click the policy, and then click Unbind Policy.
4. Click Apply Changes, and then click Close.

Configuring TCP Compression Policies

Dec 16, 2013

You can configure TCP compression policies for an application to increase the performance of the application. TCP compression reduces network latency, reduces bandwidth requirements, and increases the speed of transmission. When configuring a TCP compression policy, you associate a compression action with the policy. The compression action specifies either Compress, GZIP, Deflate, or NoCompress as the compression type. For more information about the compression policies, and compression actions, see NetScaler Gateway , Enterprise Edition at http://edocs.citrix.com/.

To configure a TCP compression policy for a resource in the NetScaler Gateway Applications node

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, in the TCP Compression column, click the icon for the application, file share, intranet subnet, or resource for which you want to configure a TCP compression policy.
3. In the Configure TCP Compression Policies dialog box, do the following:
   - To specify an existing TCP compression policy, click Insert Policy, and then, in the Policy Name column, click the name of the policy.
   - To create a new TCP compression policy, click Insert Policy, and then, in the Policy Name column, click New Policy. In the Create TCP Compression Policy dialog box, in the Policy Name box, after the underscore ("_"), type a name for the policy. Then, in Action, either select an existing action or click New and configure a new action. You can also click View to view the configured compression type.

     For more information about configuring a TCP compression policy or action, see NetScaler Gateway , Enterprise Edition at http://edocs.citrix.com/.

   - To modify a policy that you have inserted, in the Policy Name column, click the policy name, and then click Modify Policy.
   - To regenerate the priorities assigned to the policies, click Regenerate Priorities.
   - To specify a new priority value for a policy, in the Priority column, double-click the assigned priority, and then enter the value you want.
   - To unbind a policy, click the policy, and then click Unbind Policy.
4. Click Apply Changes, and then click Close.

Configuring Bookmarks

Dec 16, 2013

You can configure bookmarks for an application or for a resource that you configure in the Other Resources category if you want the application or resource to be accessible from the NetScaler Gateway home page.

To configure a bookmark for an NetScaler Gateway application or a resource in the Other Resources category

1. In the navigation pane of the NetScaler configuration utility, expand AppExpert, and then click Access Gateway Applications.
2. In the details pane, click the application or resource for which you want to configure a bookmark, and then click Configure Bookmark.
3. In the Create Bookmark dialog box, configure values for the parameters.
   For more information about the parameters in the Create Bookmark dialog box, see NetScaler Gateway , Enterprise Edition at http://edocs.citrix.com/.
4. Click Create, and then click Close.

AppQoE

Jan 07, 2014

Application level Quality of Experience (AppQoE) integrates several existing policy-based security features of the NetScaler appliance into a single integrated feature that takes advantage of a new queuing mechanism, fair queuing. Fair queuing manages requests to load-balanced web servers and applications at the virtual server level instead of at the service level, allowing it to handle queuing of all requests to a web site or application as one group before load balancing, instead of as separate streams after load balancing.

The features that are integrated into AppQoE are HTTP Denial-of-Service Protection (HDOSP), Priority Queuing (PQ), and SureConnect. Collectively these services provide protection against a number of problems:

- **Simple overload.** Any server, no matter how robust, can accept only a limited number of connections at one time. When a protected web site or application receives too many requests at once, the Surge Protection feature detects the overload and queues the excess connections til the server can accept them. The Priority Queuing feature ensures that whoever most needs access to a resource is provided access without having to wait behind other lower-priority requests. The SureConnect feature displays an alternate web page that notifies users that the resource that they requested is not available.
- **Denial-of-Service (DOS) attacks.** Any public-facing resource is vulnerable to attacks whose purpose is to bring that service down and deny legitimate users access to it. The Surge Protection, Priority Queuing, and SureConnect features help manage DOS attacks as well as other types of high load. In addition, the HTTP Denial-of-Service Protection feature targets DOS attacks against your web sites, sending challenges to suspected attackers and dropping connections if the clients do not send an appropriate response.

Until the current version of the NetScaler operating system, these features were implemented at the service level, which means that each service was assigned its own queues. While service-level queues work, they also have some disadvantages, most of which are due to the NetScaler appliance having to load balance requests before implementing any of the protection features that rely on queuing. Implementing protection features before queuing has a number of advantages, some of which are listed below:

- Absolute priority of connections as configured in the priority queuing feature can be maintained.
- Connections are not flushed if a service transitions state, as they are in a service-level queue.
- During periods of high load, such as a denial-of-service attack, HTTP DoS and SureConnect come into play before load balancing, allowing these features to detect and divert unwanted or lower-priority traffic from the load balancer before the load balancer must cope with it.

In addition to implementing fair queuing, AppQoE integrates a set of features that each provide a different set of tools to achieve a common goal: protecting your networked resources from excessive or inappropriate demand. Putting these features into a common framework enables you to configure and implement them more easily.

Enabling AppQoE

Apr 20, 2013

To configure AppQoE, you must first enable the feature.

To enable AppQoE by using the command line

At the command prompt, type the following commands:

- enable ns feature appqoe
- show ns feature

Example

```
> enable ns feature appqoe
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection          SP              ON
 3)    Load Balancing           LB              ON
...
29)   AppQoE                   AppQoE           ON
 Done
```

To enable AppQoE by using the configuration utility

1. Navigate to System > Settings.
2. In the details pane, click Configure Advanced Features.
3. In the Configure Advanced Features dialog box, select the AppQoE check box.
4. Click OK.

AppQOE Actions

Oct 02, 2014

After enabling the AppQoE feature, you must configure one or more actions for handling requests.

Important: No specific individual parameters are required to create an action, but you must include at least one parameter or you cannot create the action.

To configure an AppQoE action by using the command line

At the command prompt, type the following commands:

- add appqoe action <name> [-priority <priority>] [-respondWith (ACS | NS) [<customfile>] [-altContentSvcName <string>] [-altContentPath <string>] [-maxConn <positive_integer>] [-delay <usecs>] [-polqDepth <positive_integer>] [-priqDepth <positive_integer>] [-dosTrigExpression <expression>] [-dosAction (SimpleResponse | HICResponse)]
- show appqoe action

Example

To configure priority queuing with policy queue depths of 10 and 1000 for medium and lowest priority queues, respectively:

```
> add appqoe action appqoe-act-basic-prhigh -priority HIGH
 Done

> add appqoe action appqoe-act-basic-prmedium -priority MEDIUM -polqDepth 10
 Done

> add appqoe action appqoe-act-basic-prlow -priority LOW -polqDepth 1000
 Done

> show appqoe action
1)     Name: appqoe-act-basic-prhigh
       ActionType: PRIORITY_QUEUING
       Priority: HIGH
       PolicyQdepth: 0
       Qdepth: 0

2)     Name: appqoe-act-basic-prmedium
       ActionType: PRIORITY_QUEUING
       Priority: MEDIUM
       PolicyQdepth: 10
       Qdepth: 0

3)     Name: appqoe-act-basic-prlow
       ActionType: PRIORITY_QUEUING
       Priority: LOW
       PolicyQdepth: 1000
       Qdepth: 0
Done
```

To modify an existing AppQoE action by using the command line

At the command prompt, type the following commands:

- set appqoe action <name> [-priority <priority>] [-altContentSvcName <string>] [-altContentPath <string>] [-polqDepth <positive_integer>] [-priqDepth <positive_integer>] [-maxConn <positive_integer>] [-delay <usecs>] [-dosTrigExpression <expression>] [-dosAction (SimpleResponse | HICResponse)]
- show appqoe action

To remove an AppQoE action by using the command line

At the command prompt, type the following commands:

- rm appqoe action <name>
- show appqoe action

Parameters for configuring an AppQoE action

name

A name for the new action, or the name of the existing action that you want to modify. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols.

priority

The priority queue to which the request is assigned. When a protected web server or application is heavily loaded and cannot accept additional requests, specifies the order in which waiting requests are to be fulfilled when resources are available. The choices are:

1. **HIGH.** Fulfills the request as soon as resources are available.
2. **MEDIUM.** Fulfills the request after it has fulfilled all requests in the HIGH priority queue.
3. **LOW.** Fulfills the request after it has fulfilled all requests in the HIGH and MEDIUM priority queues.
4. **LOWEST.** Fulfills the request only after it has fulfilled all requests in higher-priority queues.

If priority is not configured, then the NetScaler appliance assigns the request to the LOWEST priority queue by default.

respondWith

Configures the NetScaler ADC to take the specified Responder action when the specified threshold is reached. Must be used with one of the following settings:

- **ACS:** Serves content from an alternate content service. Threshold: maxConn (maximum connections) or delay.
- **NS:** Serves a built-in response from the NetScaler ADC. Threshold: maxConn (maximum connections) or delay.
- **NO ACTION:** Serves no alternative content. Assigns connections to the LOWEST priority queue if the maxConn (maximum connections) or delay threshold is reached.

altContentSvcName

If -responseWith ACS is specified, the name of the alternative content service, usually an absolute URL to the web server that hosts the alternate content.

altContentPath

If -responseWith (ACS | NS) is specified, the path to the alternative content.

polqDepth

Policy queue depth threshold value for the policy queue associated with this action. When the number of connections in the policy queue associated with this action increases to the specified number, subsequent requests are assigned to the LOWEST policy queue. Minimum value: 1 Maximum value: 4,294,967,294

priqDepth

Policy queue depth threshold value for the specified priority queue. If the number of requests in the specified queue on the virtual server to which the policy associated with the current action is bound increases to the specified number, subsequent requests are assigned to the LOWEST priority queue. Minimum value: 1 Maximum value: 4,294,967,294

maxConn

The maximum number of connections that can be open for requests that match the policy rule. Minimum value: 1 Maximum value: 4,294,967,294

delay

The delay threshold, in microseconds, for requests that match the policy rule. If a matching request has been delayed for longer than the threshold, the NetScaler appliance performs the specified action. If NO ACTION is specified, then the appliance assigns requests to the LOWEST priority queue. Minimum value: 1 Maximum value: 599999,999

dosTrigExpression

Adds an optional second-level check to trigger DoS actions.

dosAction

Action to take when the ADC determines that it or a protected server is under DoS attack. Possible values: SimpleResponse, HICResponse

To configure an AppQoE action by using the configuration utility

1. Navigate to App-Expert > AppQoE > Actions.
2. In the details pane, do one of the following:
   - To create a new action, click Add.
   - To modify an existing action, select the action, and then click Edit.
3. In the Create AppQoE Action or the Configure AppQoE Action screen, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring the AppQoE Action" as follows (asterisk indicates a required parameter):
   - Name—name
   - Action type—respondWith
   - Priority—priority
   - Policy Queue Depth—polqDepth
   - Queue Depth—priqDepth
   - DOS Action—dosAction
4. Click Create or OK.

AppQoE Parameters

Oct 02, 2014

In the AppQoE parameters, you configure the session life of an AppQoE session, the file name of the file containing the customized response, and the number of client connections that can be placed in a queue.

To configure the AppQoE parameter settings by using the command line

At the command prompt, type the following commands:

- set appqoe parameter [-sessionLife <secs>] [-avgwaitingclient <positive_integer>] [-MaxAltRespBandWidth <positive_integer>] [-dosAttackThresh <positive_integer>]
- show appqoe parameter

Parameters for configuring the AppQoE parameters

sessionLife

Number of seconds to wait after displaying alternate content before the ADC displays the same content again. Default value: 300 Maximum Minimum value: 1 Maximum value: 4,294,967,294

avgwaitingclient

The average number of client requests that can be in the service waiting queue. Default value: 1000000 Maximum value: 4,294,967,294

MaxAltRespBandWidth

The maximum bandwidth to consume when sending alternate responses. If the maximum is reached, the ADC quits sending the alternate content til bandwidth consumption drops. Default value: 100 Minimum value: 1 Maximum value: 4,294,967,294

dosAtckThrsh

The denial-of-service attack threshold. The number of connections that must be waiting in queues before the ADC responds with DoS protection measures. Default value: 2000 Minimum value: 0 Maximum value: 4,294,967,294

To configure the AppQoE parameter settings by using the configuration utility

1. Navigate to AppExpert > AppQoE.
2. In the details pane, click Configure AppQoE Parameters.
3. In the Configure AppQoE params screen, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring the AppQoE Parameters" as follows (asterisk indicates a required parameter):
   - Session Life (secs)—sessionLife
   - Average waiting client—avgwaitingclient
   - Alternate Response Bandwidth Limit(Mbps) —MaxAltRespBandWidth
   - DOS Attack Threshold —dosAttackThresh
4. Click OK.

AppQoE Policies

Oct 02, 2014

To implement AppQoE, you must configure at least one policy to tell your NetScaler ADC how to distinguish the connections to be queued in a specific queue.

To configure an AppQoE policy by using the command line

At the command prompt, type the following command:
add appqoe policy <name> -rule <expression> -action <string>
Example

The following example selects requests with a User-Agent header that contains "Android", and assigns them to the medium priority queue. These requests come from smartphones and tablets that run the Google Android operating system.

```
> add appqoe action appqoe-act-primd -priority MEDIUM
Done
> add appqoe policy appqoe-pol-primd -rule "HTTP.REQ.HEADER(\"User-Agent\").CONTAINS(\"Android\")" -action appqoe-act-primd
Done
> sh appqoe policy appqoe-pol-primd
      Name: appqoe-pol-primd
      Rule: HTTP.REQ.HEADER("User-Agent").CONTAINS("Android")
      Action: appqoe-act-primd
      Hits: 0

Done
```

Parameters for configuring an AppQoE policy

name

A name for the AppQoE policy. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. You should chose a name that helps identify the type of action.

rule

A NetScaler expression that tells the appliance which connections it should handle. For complete information about policy expressions, see the *Citrix NetScaler Policy Configuration and Reference Guide* at .

action

The AppQoE action to perform when a connection matches the policy.

To configure an AppQoE policy by using the configuration utility

1. Navigate to App-Expert > AppQoE > Policies.
2. In the details pane, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the policy, and then click Edit.
3. If you are creating a new policy, in the Create AppQoE Policy dialog, in the Name text box, type a name for your new policy.
   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. You should chose a name that helps identify the purpose and effect of this policy.

   If you are modifying an existing policy, skip this step. You cannot change the name of an existing policy.

4. In the Action drop-down list, choose the AppQoE action to perform when the policy matches a connection. Click the plus (+) to open the Add AppQoE Action dialog and add a new action.
5. In the Rule text box, either enter the policy expression directly, or click New to create a policy expression. If you click New, perform the following steps:
   1. In the Create Expression dialog box, click Add.
   2. In the Add Expression dialog box, select a common expression from the Frequently Used Expressions drop-down list, or use the Construct Expression drop-down lists to create the expression that defines which traffic to filter.
      If you choose to create your own expression, you start by selecting the first term from the first drop-down list on the left side of the Construct

Expression area. The choices in that list are:

- HTTP: All traffic to port 80 and port 443.
- SYS:
- CLIENT:
- SERVER:
- ANALYTICS:
- TEXT:

The default choice is HTTP. After you make a choice in the first drop-down list (or accept the default), you can choose the next term in your expression from the drop-down list to the right of it. The terms in that list and other lists that follow change depending on your previous choices; the lists offer only terms that are valid choices. Continue to select terms until you have finished the expression.

Use the Help and Preview Expression areas for assistance when creating the expression. For a complete description of the available choices, see the *Citrix NetScaler Policy Configuration and Reference Guide* at .

3. When you have created the expression that you want, click OK. The expression is added in the Expression text box.

6. Click Create. The expression appears in the Rule text box.

Entity Templates

May 21, 2015

An entity template is a collection of configuration information for an individual entity on a Citrix® NetScaler® appliance. It provides a specification and a set of defaults for a configurable NetScaler entity, such as a policy, virtual server, service, or action. By using a template that defines a set of defaults, you can quickly configure multiple entities that require a similar configuration while eliminating several configuration steps.

Entity templates are available only in the configuration utility. You use the NetScaler configuration utility to create, manage, and use any type of entity template. You can share entity templates with other administrators and manage local folders that contain the templates. You can also import entity templates from and export entity templates to your local computer.

Before creating a template, you should be familiar with the configuration of the entity.

Note: You use entity templates to configure individual entities. To configure multiple entities related to a particular Web application, you must use an application template. For more information, see "AppExpert Applications and Templates."

How Entity Templates Work

When you create a template for a NetScaler entity, you specify default values for the entity. You specify what values must be read-only, what values must not be displayed, and what values users can configure. You also configure the pages that compose the template import wizard. All the information and settings you provide are stored in the template file.

When a user imports the entity template to a NetScaler appliance, a wizard guides the user through the various pages that you configured for the template. The wizard displays the read-only parameter values and prompts the user to specify values for the configurable parameters. After the user follows the instructions in the wizard, the appliance creates the entity with the configured values.

For example, you can create an entity template for HTTP services that provides a text box for a service name and assigns preset values for the service protocol, timeouts, thresholds, and monitors. Later, when you use the template to create new HTTP services, a wizard prompts you for a service name and supplies the preset values that you would otherwise have configured manually.

The procedure for creating entity templates for load balancing virtual servers is different than the AppExpert procedure for creating other entity templates. For more information, see "Creating an Entity Template."

In addition, the procedure for using the template to create the load balancing virtual server entity is different. For more information, see "Creating an Entity from a Template."

Configuring an Entity Template

Mar 28, 2012

You can create or modify an entity template either from the AppExpert feature node or from the associated NetScaler feature node for the entity. For example, you can create a content switching virtual server entity template in either the AppExpert feature node or the content switching feature node in the configuration utility.

If you create a template that is not based on an existing entity, you can specify the following options and settings for the template:

- The default value of a parameter.
- Whether the default values are visible to users.
- Whether the default values can be changed by users.
- The number of pages in the entity import wizard, including the page names, text, and available parameters.
- The entities that must be bound to the entity for which the template is being created.

  For example, when you are creating a cache redirection virtual server template, you can specify the policies that you want to bind to the cache redirection virtual servers that you create from the template. However, only binding information is included in the template. The bound entities are not included. If the entity template is imported to another NetScaler appliance, the bound entities must exist on the appliance at import time for the binding to succeed. If none of the bound entities exist on the target appliance, the entity (for which the template was configured) is created without any bindings. If only a subset of the bound entities exist on the target appliance, they are bound to the entity that is created from the template.

When you create a template based on an existing entity, the configuration settings of the entity appear in the template. All bound entities are selected by default, but you can modify bindings as necessary. As in the case of a template that is not based on an existing entity, only binding information is included and not the entities. You can either save the template with the existing configuration settings or use the settings as a basis for creating a new configuration for a template.

Creating an Entity Template

Aug 30, 2013

You can create entity templates in either the AppExpert node or the NetScaler feature node that corresponds to the type of entity. For example, you can create a content switching virtual server template from the entity templates tab of the AppExpert feature's Templates node or in the Content Switching node. You can also specify the parameters that you want the template to store, and specify whether you want the template import wizard to prompt the user for certain parameter values.

However, when creating load balancing virtual servers, you do not have the option of specifying parameter values that you want stored in the template. You create a load balancing virtual server template by selecting an existing load balancing virtual server and configuring any variables that you might want to create in existing parameters and bound policies. The variables can be assigned values when you create a load balancing virtual server from the template. The template stores load balancing parameters such as the virtual server's IP address and port number, bound policies, actions, and variable definitions. A deployment file is also created, automatically, from the load balancing configuration. The deployment file stores deployment-specific information, such as information about bound services, service groups, and the name-value pairs of variables. If the bound entities that are included in the template are already configured on the NetScaler appliance to which the template is imported, duplicates are created, with names that are generated automatically in a particular format. The duplicate entities are based on the parameter information stored in the entity template.

When you create a load balancing virtual server template from the AppExpert node, the template is always saved to the /nsconfig/nstemplates/entities/lb vserver/ folder. If you want to save the template to a different folder, create the template from the Virtual Servers pane in the Load Balancing node. The deployment file is created with the name with which you save the template file, but with the string _deployment appended to the name. The deployment file is saved to the /nsconfig/nstemplates/entities/lb vserver/deployment_files/ folder. For more information about deployment files for load balancing virtual server templates, see "Understanding Load Balancing Entity Templates and Deployment Files."

Note: You can use either of the first two procedures for creating any template, except for a load balancing virtual server template. For creating a load balancing virtual server template, use the third or fourth procedure.

To create an entity template by using the AppExpert feature node

1. Navigate to AppExpert > Templates.
2. In the details pane, on the Entity Templates tab, do one of the following:
   - To create a new template, click Add. In the Select the Template Type dialog box, select the template type, and then click OK.
   - To create a duplicate of an existing entity template, in the details pane, select the entity template, and then click Add.
3. In the Create...Template dialog box, follow the instructions to create a template.
   If you are creating a duplicate of an existing entity template, in the Create...Template dialog box, on the Specify Template Name page, you must change the name of the entity template.
4. Click Finish, and then click Exit.

To create an entity template by using its corresponding feature node

1. Navigate to Traffic Management, and select the feature (for example, Content Switching), and then select the entity (for example, Virtual Servers), for which you want to create the entity template.
2. At the top of the details pane, click Entity Templates, and then click Create Template.

3. In the Create...Template dialog box, follow the instructions to create a template.
4. Click Finish, and then click Exit.

To create a load balancing virtual server template from the AppExpert node

1. Navigate to AppExpert > Templates.
2. In the details pane, on the LB Templates tab, click Add.
3. In the Select Load Balancing Virtual Server dialog box, select the load balancing virtual server whose configuration you want to save to a template file, and then click OK.
4. In the Create Template dialog box, provide the following information:
   - Name. The name of the template.
     Note: The Folder field shows the location to which the template will be saved. You cannot modify the path that is displayed.
   - Configure Variables. Configure variables for the load balancing template. For more information, see "Configuring Variables in Load Balancing Virtual Server Templates."
   - Introduction Description. A description of the virtual server for which you are creating a template.
   - Summary Description. A summary of the configuration or additional instructions for other administrators, such as a description of any additional steps that need to be followed after the entity is successfully created.
   - Author. The creator of the template.
   - Major. An optional major version number of your choice, to be specified if you want to maintain versions of your template.
   - Minor. An optional minor version number of your choice, to be specified if you want to maintain minor versions of your template.
     You can maintain versions by incrementing one or both of the version numbers each time you maintain the template. The Entity Template Wizard concatenates and displays the major and minor version numbers during import. For example, if the major version number is 1 and the minor version is 1, the Entity Template Wizard displays a version number of 1.1.
5. Click OK.

To create a load balancing virtual server template from the Load Balancing Virtual Servers pane

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server on which to base the template,, and then click Create Template. You might have to click the scroll arrow at the bottom right of the pane to bring the Create Template button into view.
3. In the Create Template dialog box, provide the following information:
   - Name. The name of the template.
   - Folder. The location to which the template will be saved.
     Note: If you want to save the template to the appliance, you can save it only to the `/nsconfig/nstemplates/entities/lb vserver/` directory (the path displayed by default in Folder. If you want to save the template file to a folder on your computer, click the down-arrow on the Browse button, click Local, and then select a folder.
   - Configure Variables. Configure variables for the load balancing template. For more information, see "Configuring Variables in Load Balancing Virtual Server Templates."
   - Introduction Description. A description of the virtual server for which you are creating a template.
   - Summary Description. A summary of the configuration or additional instructions for other administrators, such as a description of any additional steps that need to be followed after the entity is successfully created.
   - Author. The creator of the template.

- Major. An optional major version number of your choice, to be specified if you want to maintain versions of your template.
- Minor. An optional minor version number of your choice, to be specified if you want to maintain minor versions of your template.
  You can maintain versions by incrementing one or both of the version numbers each time you maintain the template. The Entity Template Wizard concatenates and displays the major and minor version numbers during import. For example, if the major version number is 1 and the minor version is 1, the Entity Template Wizard displays a version number of 1.1.

4. Click OK.

Configuring Variables in Load Balancing Virtual Server Templates

Aug 30, 2013

Load balancing virtual server templates support the declaration of variables in the configured load balancing parameters and in bound policies and actions. The ability to declare variables enables you to replace preconfigured values with values that suit the environment into which you are importing the template. The Entity Template Wizard, which appears when you import a template, includes a Specify Variable Values page on which you can specify appropriate values for the variables that are configured for the entity template. This wizard page appears only when you import a template that is configured with existing variables.

As an example, consider the following expression configured for a policy that is bound to a load balancing virtual server for which you are creating a template. The expression evaluates the value of the Accept-Language header in an HTTP request.

```
HTTP.REQ.HEADER("Accept-Language").CONTAINS("en-us")
```

If you want the value of the header to be configurable at import time, you can specify the string en-us as a variable. When importing the template, you can specify a new value for the variable on the Specify Variable Values page.

After you create a variable, you can do the following:

- Assign additional strings to an existing variable. After you create a variable for a string, you can select and assign other parts of the same or different expression to the variable. The strings you assign to a variable need not be the same. At import time, all the strings that are assigned to the variable are replaced with the value that you provide.
- View the string or strings that are assigned to the variable.
- View a list of all the entities and parameters that use the variable.

To configure variables in a load balancing virtual server template

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, right-click the virtual server that you want to export to a template file, and then click Create Template.
3. In the Create Template dialog box, modify the default template file name if required, specify the location where you want to save the template, and then click Configure Variables.
4. In the Configure Variables dialog box, click the tab that lists the entity for which you want to configure a variable, select the entity, and then click Configure Variables.
5. In the Variables for <Entity Type>: <Entity Name> dialog box, click the  button next to the parameter value or expression in which you want to create a variable.
6. In the Variables for <Field Name> dialog box, do the following:
   - To create a variable, in the text box that displays the configured expression or value, select the string that you want to be configurable at import time, and then click Add. In the Create Variable dialog box, specify a name and a description for the variable, and then click Create.

     The name of the variable, its value, and the description you provided appear in the Available Variables listing in the dialog box. The name you provide will be the name of the associated field in the template import wizard, and the description will appear as alt text when the user positions the mouse pointer over the field.

   - To modify a variable, in the Available Variables list, click the variable, and then click Open. In the Create Variable dialog

box, modify the value and the description, and then click OK.

The new value that you specify will not replace the text selected in the text box that displays the configured expression or value. However, when you import the template, the new value will be displayed as the default value for the variable in the template import wizard.

- To view all the strings that are assigned to a given variable, in the Available Variables listing, click the name of the variable. The strings that are assigned to the variable are highlighted.
- To view a list of all the parameters, expressions, and actions in which the variable is used, in the Available Variables listing, click the variable whose references you want to view, and then click Show References.
- To assign a string to an existing variable, in the text box that displays the expression you configured, select the string you want to assign to an existing variable, right-click the selection, click Use existing Variable, and then click the name of the variable to which you want to assign the string.

  If a variable has multiple strings assigned to it, when you specify a new value for the variable during import, all strings assigned to the variable are replaced with the new value.

7. Click Close.

Modifying an Entity Template

Aug 30, 2013

You can modify only the parameters, bindings, and pages configured for a template. The name and location of the template specified when the template was created cannot be changed. The NetScaler appliance does not provide you with the option of modifying a load balancing virtual server template.

To modify an entity template by using the AppExpert feature node

1. Navigate to AppExpert > Templates.
2. In the details pane, on the Entity Templates tab, select the template you want to change, and then click Open.
3. In the Modify...Template dialog box, follow the instructions to modify a template.
4. Click Finish, and then click Exit.

To modify an entity template by using its corresponding feature node

1. Navigate to Traffic Management, select the feature (for example, Content Switching), and then select the entity (for example, Virtual Servers) for which you want to modify the entity template.
2. At the top of the details pane, click Entity Templates, and then click Manage Template.
3. In the Manage <feature entity name> Entity Templates dialog box, select the template that you want to modify, and then click Modify.
4. In the Modify <template name> Template dialog box, follow the instructions to modify a template.
5. Click Finish, and then click Exit.
6. Click Close.

Deleting an Entity Template

Aug 30, 2013

Deleting an entity template does not affect any objects that have been created by using the template. You can delete a load balancing virtual server template only from the AppExpert feature node.

To delete an entity template by using the AppExpert feature node

1. Navigate to AppExpert > Templates.
2. In the details pane, on the Entity Templates tab, click the template you want to delete, and then click Remove.

To delete an entity template by using its corresponding feature node

1. Navigate to Traffic Management, and select the feature (for example, Content Switching) and then select the entity (for example, Virtual Servers), for which you want to delete the entity template.
2. At the top of the details pane, click Entity Templates, and then click Manage Template.
3. In the Manage...Entity Templates dialog box, select the template that you want to delete, and then click Delete.

Creating an Entity from a Template

Aug 30, 2013

You can create an entity from an entity template either from the AppExpert feature node in the NetScaler configuration utility or from the NetScaler feature node that corresponds to the type of entity that you want to create. For example, you can create a content switching virtual server from a template with either the AppExpert feature node or the content switching feature node in the configuration utility.

The procedure for creating a load balancing virtual server from a template is different than the AppExpert procedure for creating other entities from templates.

After you create an instance of an entity using an entity template, you can configure it in the same way that you would any other object of that type, such as by using the configuration utility or the command line.

To create an entity from a template by using the AppExpert feature node

1. Navigate to AppExpert > Templates.
2. In the details pane, do one of the following:
   1. To create any entity other than a load balancing virtual server from a template, on the Entity Templates tab, click the template that you want to use, and then click Use Template.
   2. To create a load balancing virtual server from a template, on the LB Templates tab, click the template that you want to use, and then click Use Template.
3. In the <Entity Template Name> wizard, follow the instructions to create the entity on the NetScaler.
4. Click Finish, and then click Exit.

To create an entity from a template by using its corresponding feature node

1. Navigate to Traffic Management, and expand a feature node (for example, Content Switching), and then click an entity subnode (for example, Virtual Servers).
2. At the top of the details pane, click Entity Templates, and then click Use Template.
3. Click the name of the template that you want to use.
4. In the Use <template name> Template wizard, follow the instructions to create the entity.
   Only templates that match the current context are displayed. For example, in the details pane for content switching virtual servers, only entity templates for content switching virtual servers appear, if configured.
5. Click Finish, and then click Exit.

To create a load balancing virtual server by using a load balancing virtual server template

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Use Template.
3. In the Entity Template Wizard, follow the instructions to create a load balancing virtual server on the NetScaler.
   Only templates that match the current context are displayed. For example, when you click Browse (Appliance), only entity templates for load balancing virtual servers appear, if configured.
4. Click Finish, and then click Exit.
   Note: The Entity Template Wizard includes a Specify Variable Values page on which you can specify new values for variables. For more information about configuring variables in load balancing virtual server templates, see "Configuring Variables in Load Balancing Virtual Server Templates."

Managing Entity Template Folders

Aug 30, 2013

You can organize only load balancing virtual server template folders.

To organize load balancing virtual server template folders

1. Navigate to AppExpert > Templates > LB Templates.
2. In the Manage LB Templates dialog box, do one of the following:
   - To change the name of a folder, select the folder and click Rename.

     You can also click the folder that you want to rename, and then press F2. You cannot rename the top-level default folder.

   - To remove the folder, select the folder and click Delete.

     You can also click the folder that you want to remove, and then press the Delete key. You cannot remove the top-level default folder.
3. Click Close.

Uploading and Downloading Entity Templates

Aug 30, 2013

You can import the entity templates that are stored on your local computer. You can also download entity templates from the NetScaler appliance to your local computer and then import them to other NetScaler appliances.

Note: You cannot upload or download load balancing virtual server templates.

To upload an entity template to the NetScaler appliance

1. Navigate to Traffic Management, and expand a feature node (for example, Content Switching), and then click a subnode (for example, Virtual Servers) for which you want to upload an entity template.
2. At the top of the details pane, click Entity Templates, and then click Manage Template.
3. In the Manage...Entity Templates dialog box, click the top-level folder, and then click Upload.
4. In the Upload Entity Template dialog box, navigate to the template file that you want to upload, and then click Select.
5. Click Close.

To download an entity template from the NetScaler appliance

1. Navigate to Traffic Management, and expand a feature node (for example, Content Switching), and then click a subnode (for example, Virtual Servers) for which you want to upload an entity template.
2. At the top of the details pane, click Entity Templates, and then click Manage Template.
3. In the Manage...Entity Templates dialog box, click the template that you want to download, and then click Download.
4. In the Download Entity Template dialog box, navigate to the location at which you want to save the template on your local computer, enter a file name, and then click Save.
5. Click Close.

Understanding Load Balancing Entity Templates and Deployment Files

Mar 28, 2012

Load balancing entity templates are created in the same way that NetScaler application templates are created. When you export a load balancing virtual server to a template file, the following two files are automatically created:

- **Load balancing virtual server template file.** Contains XML elements that store the values of the parameters that are configured for the load balancing virtual server. The file also contains XML elements for storing information about bound policies.
- **Deployment file.** Contains XML elements that store deployment-specific information such as services, service groups, and configured variables.

In the template and deployment files, each unit of configuration information is encapsulated in a specific XML element that is meant for that unit type. For example, the load balancing method parameter, lbMethod, is encapsulated within the <lbmethod> and </lbmethod> tags.

Note: After you export a load balancing virtual server, you can add elements, remove elements, and modify existing elements before importing the configuration information to a NetScaler appliance.

Example of a Load Balancing Virtual Server Template

Following is an example of a template file that was created from a load balancing virtual server called "Lbvip":

```
<?xml version="1.0" encoding="UTF-8" ?>
 <template>
  <template_info>
   <entity_name>Lbvip</entity_name>
   <version_major>10</version_major>
   <version_minor>0</version_minor>
   <build_number>40.406</build_number>
  </template_info>
  <entitytemplate>
   <lbvserver_list>
    <lbvserver>
     <name>Lbvip</name>
     <servicetype>HTTP</servicetype>
     <ipv46>0.0.0.0</ipv46>
     <ipmask>*</ipmask>
     <port>0</port>
     <range>1</range>
     <persistencetype>NONE</persistencetype>
     <timeout>2</timeout>
     <persistencebackup>NONE</persistencebackup>
     <backuppersistencetimeout>2</backuppersistencetimeout>
     <lbmethod>LEASTCONNECTION</lbmethod>
     <persistmask>255.255.255.255</persistmask>
     <v6persistmasklen>128</v6persistmasklen>
     <pq>OFF</pq>
```

```
        <sc>OFF</sc>
        <m>IP</m>
        <datalength>0</datalength>
        <dataoffset>0</dataoffset>
        <sessionless>DISABLED</sessionless>
        <state>ENABLED</state>
        <connfailover>DISABLED</connfailover>
        <clttimeout>180</clttimeout>
        <somethod>NONE</somethod>
        <sopersistence>DISABLED</sopersistence>
        <sopersistencetimeout>2</sopersistencetimeout>
        <redirectportrewrite>DISABLED</redirectportrewrite>
        <downstateflush>DISABLED</downstateflush>
        <gt2gb>DISABLED</gt2gb>
        <ipmapping>0.0.0.0</ipmapping>
        <disableprimaryondown>DISABLED</disableprimaryondown>
        <insertvserveripport>OFF</insertvserveripport>
        <authentication>OFF</authentication>
        <authn401>OFF</authn401>
        <push>DISABLED</push>
        <pushlabel>none</pushlabel>
        <l2conn>OFF</l2conn>
        <appflowlog>DISABLED</appflowlog>
        <icmpvsrresponse>PASSIVE</icmpvsrresponse>
        <lbvserver_cmppolicy_binding_list>
          <lbvserver_cmppolicy_binding>
            <name>Lbvip</name>
            <policyname>NOPOLICY-COMPRESSION</policyname>
            <priority>100</priority>
            <gotopriorityexpression>END</gotopriorityexpression>
            <bindpoint>REQUEST</bindpoint>
          </lbvserver_cmppolicy_binding>
        </lbvserver_cmppolicy_binding_list>
      </lbvserver>
    </lbvserver_list>
  </entitytemplate>
 </template>
```

Example of a Deployment File

Following is the deployment file associated with the virtual server in the preceding example:

```
<?xml version="1.0" encoding="UTF-8" ?>
 <template_deployment>
  <template_info>
   <entity_name>Lbvip</entity_name>
   <version_major>10</version_major>
   <version_minor>0</version_minor>
   <build_number>40.406</build_number>
  </template_info>
```

```
  <service_list>
   <service>
   <ip>1.2.3.4</ip>
   <port>80</port>
   <servicetype>HTTP</servicetype>
   </service>
  </service_list>
  <servicegroup_list>
   <servicegroup>
    <name>svcgrp</name>
    <servicetype>HTTP</servicetype>
    <servicegroup_servicegroupmember_binding_list>
     <servicegroup_servicegroupmember_binding>
      <ip>1.2.3.90</ip>
      <port>80</port>
     </servicegroup_servicegroupmember_binding>
    <servicegroup_servicegroupmember_binding>
     <ip>1.2.8.0</ip>
     <port>80</port>
    </servicegroup_servicegroupmember_binding>
    <servicegroup_servicegroupmember_binding>
     <ip>1.2.8.1</ip>
     <port>80</port>
    </servicegroup_servicegroupmember_binding>
    <servicegroup_servicegroupmember_binding>
     <ip>1.2.9.0</ip>
     <port>80</port>
    </servicegroup_servicegroupmember_binding>
   </servicegroup_servicegroupmember_binding_list>
  </servicegroup>
 </servicegroup_list>
</template_deployment>
```

HTTP Callouts

Jan 11, 2013

For certain types of requests, or when certain criteria are met during policy evaluation, you might want to stall policy evaluation briefly, retrieve information from a server, and then perform a specific action that depends on the information that is retrieved. At other times, when you receive certain types of requests, you might want to update a database or the content hosted on a Web server. HTTP callouts enable you to perform all these tasks.

An HTTP callout is an HTTP or HTTPS request that the NetScaler appliance generates and sends to an external application when certain criteria are met during policy evaluation. The information that is retrieved from the server can be analyzed by default syntax policy expressions, and an appropriate action can be performed. You can configure HTTP callouts for HTTP content switching, TCP content switching, rewrite, responder, and for the token-based method of load balancing.

Before you configure an HTTP callout, you must set up an application on the server to which the callout will be sent. The application, which is called the *HTTP callout agent* , must be configured to respond to the HTTP callout request with the required information. The HTTP callout agent can also be a Web server that serves the data for which the NetScaler appliance sends the callout. You must make sure that the format of the response to an HTTP callout does not change from one invocation to another.

After you set up the HTTP callout agent, you configure the HTTP callout on the NetScaler appliance. Finally, to invoke the callout, you include the callout in a default syntax policy in the appropriate NetScaler feature and then bind the policy to the bind point at which you want the policy to be evaluated.

After you have configured the HTTP callout, you must verify the configuration to make sure that the callout is working correctly.

How an HTTP Callout Works

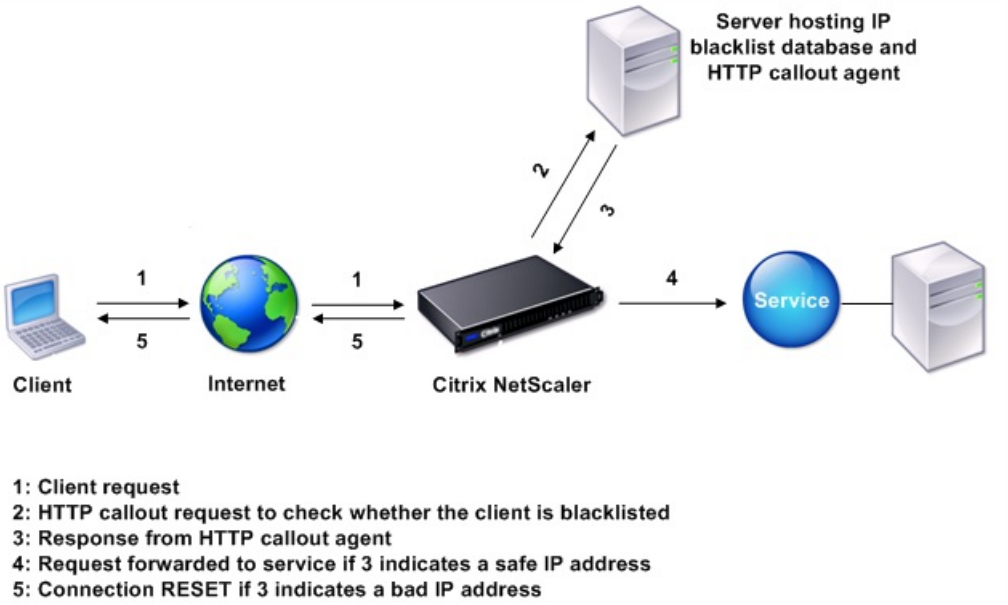
Jan 11, 2013

When the NetScaler appliance receives a client request, the appliance evaluates the request against the policies bound to various bind points. During this evaluation, if the appliance encounters the HTTP callout expression, SYS.HTTP_CALLOUT(<name>), it stalls policy evaluation briefly and sends a request to the HTTP callout agent by using the parameters configured for the specified HTTP callout. Upon receiving the response, the appliance inspects the specified portion of the response, and then either performs an action or evaluates the next policy, depending on whether the evaluation of the response from the HTTP callout agent evaluates to TRUE or FALSE, respectively. For example, if the HTTP callout is included in a responder policy, if the evaluation of the response evaluates to TRUE, the appliance performs the action associated with the responder policy.

If the HTTP callout configuration is incorrect or incomplete, or if the callout invokes itself recursively, the appliance raises an UNDEF condition, and updates the undefined hits counter.

The following figure illustrates the working of an HTTP callout that is invoked from a globally bound responder policy. The HTTP callout is configured to include the IP address of the client that is associated with an incoming request. When the NetScaler appliance receives a request from a client, the appliance generates the callout request and sends it to the callout server, which hosts a database of blacklisted IP addresses and an HTTP callout agent that checks whether the client's IP address is listed in the database. The HTTP callout agent receives the callout request, checks whether the client's IP address is listed, and sends a response that the NetScaler appliance evaluates. If the response indicates that the client's IP address is not blacklisted, the appliance forwards the response to the configured service. If the client's IP address is blacklisted, the appliance resets the client connection

Figure 1. HTTP Callout Entity Model



Notes on the Format of HTTP Requests and Responses

May 21, 2015

The NetScaler appliance does not check for the validity of the HTTP callout request. Therefore, before you configure HTTP callouts, you must know the format of an HTTP request. You must also know the format of an HTTP response, because configuring an HTTP callout involves configuring expressions that evaluate the response from the HTTP callout agent.

This document includes the following details:

- Format of an HTTP Request
- Format of an HTTP Response

Format of an HTTP Request

An HTTP request contains a series of lines that each end with a carriage return and a line feed, represented as either <CR><LF> or \r\n.

The first line of a request (the *message line*) contains the HTTP method and target. For example, a message line for a GET request contains the keyword GET and a string that represents the object that is to be fetched, as shown in the following example:

```
GET /mysite/mydirectory/index.html HTTP/1.1\r\n
```

The rest of the request contains HTTP headers, including a required Host header and, if applicable, a message body.

The request ends with a bank line (an extra <CR><LF> or \r\n).

Following is an example of a request:

```
Get /mysite/index.html HTTP/1.1\r\n
Host: 10.101.101.10\r\n
Accept: */*\r\n
\r\n
```

Format of an HTTP Response

An HTTP response contains a status message, response HTTP headers, and the requested object or, if the requested object cannot be served, an error message.

Following is an example of a response:

```
HTTP/1.1 200 OK\r\n
Content-Length: 55\r\n
Content-Type: text/html\r\n
Last-Modified: Wed, 12 Aug 1998 15:03:50 GMT\r\n
Accept-Ranges: bytes\r\n
ETag: “04f97692cbd1:377”\r\n
Date: Thu, 19 Jun 2008 19:29:07 GMT\r\n
\r\n
```

<55-character response>

Configuring an HTTP Callout

Dec 17, 2013

When configuring an HTTP callout, you specify the type of request (HTTP or HTTPS), destination and format of the request, the expected format of the response, and, finally, the portion of the response that you want to analyze.

For the destination, you either specify the IP address and port of the HTTP callout agent or engage a load balancing, content switching, or cache redirection virtual server to manage the HTTP callout requests. In the first case, the HTTP callout requests will be sent directly to the HTTP callout agent. In the second case, the HTTP callout requests will be sent to the virtual IP address (VIP) of the specified virtual server. The virtual server will then process the request in the same way as it processes a client request. For example, if you expect a large number of callouts to be generated, you can configure instances of the HTTP callout agent on multiple servers, bind these instances (as services) to a load balancing virtual server, and then specify the load balancing virtual server in the HTTP callout configuration. The load balancing virtual server then balances the load on those configured instances as determined by the load balancing algorithm.

For the format of the HTTP callout request, you can specify the individual attributes of the HTTP callout request (an attribute-based HTTP callout), or you can specify the entire HTTP callout request as a default syntax expression (an expression-based HTTP callout).
Note: The appliance does not check for the validity of the request. You must make sure that the request is a valid request. An incorrect or incomplete HTTP callout configuration results in a runtime UNDEF condition that is not associated with an action. The UNDEF condition merely updates the Undefined Hits counter, which enables you to troubleshoot an incorrectly configured HTTP callout. However, the appliance parses the HTTP callout request to enable you to configure certain NetScaler features for the callout. This can lead to an HTTP callout invoking itself. For information about callout recursion and how you can avoid it, see "Avoiding HTTP Callout Recursion."
Finally, regardless of whether you use HTTP request attributes or an expression to define the format of the HTTP callout request, you must specify the format of the response from the HTTP callout agent and the portion of the response that you want to evaluate. The response can be a Boolean value, a number, or text. The portion of the response that you want to evaluate is specified by an expression. For example, if you specify that the response contains text, you can use HTTP.RES.BODY(<unit>) to specify that the appliance must evaluate only the first <unit> bytes of the response from the callout agent.

At the command line, you first create an HTTP callout by using the add command. When you add a callout, all parameters are set to a default value of NONE, except the HTTP method, which is set to a default value of GET. You then configure the callout's parameters by using the set command. The set command is used to configure both types of callouts (attribute-based and expression-based). The difference lies in the parameters that are used for configuring the two types of callouts. Accordingly, the command-line instructions that follow include a set command for configuring an attribute-based callout and a set command for configuring an expression-based callout. In the configuration utility, all of these configuration tasks are performed in a single dialog box.
Note: Before you put an HTTP callout into a policy, you can modify all configured parameters except the return type. Once an HTTP callout is in a policy, you cannot completely modify an expression that is configured in the callout. For example, you cannot change HTTP.REQ.HEADER("myval") to CLIENT.IP.SRC. However, you can modify the operators and arguments that are passed to the expression. For example, you can change HTTP.REQ.HEADER("myVal1") to HTTP.REQ.HEADER("myVal2"), or HTTP.REQ.HEADER("myVal") to HTTP.REQ.HEADER("myVal").AFTER_STR(<string>). If the set command fails, create a new HTTP callout.
HTTP callout configuration involves configuring default syntax expressions. For more information about configuring default syntax expressions, see "Configuring Default Syntax Expressions: Getting Started."

To configure an HTTP callout by using the command line interface

At the command prompt, do the following:

1. Create a HTTP callout.
   add policy httpCallout <name>

   Example
   > add policy httpCallout mycallout
2. Configure the details of the HTTP callout.
   - To configure an attribute-based HTTP callout, type:
     set policy httpCallout <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-port <port|*>] [-vServer <string>] [-returnType <returnType>] [-httpMethod (GET | POST)] [-hostExpr <string>] [-urlStemExpr <string>] [-headers <name(value)> ...] [-parameters <name(value)> ...] [-resultExpr <string>]

Example

```
> set policy httpCallout mycallout -vserver lbv1 -returnType num -httpMethod GET -hostExpr 'http.req.header("Host")'
-urlStemExpr "http.req.url" -parameters Name("My Name") -headers Name("MyHeader")
-resultExpr "http.res.body(10000).length"
```

- To configure an expression-based HTTP callout, type:
  set policy httpCallout <name> [-vServer <string>] [-returnType <returnType>] [-httpMethod (GET | POST)] [-fullReqExpr <string>] [-resultExpr <string>]

Example

```
> set policy httpCallout mycallout1 -vserver lbv1 -returnType num -httpMethod GET
-fullReqExpr q{"GET " + http.req.url + "HTTP/" + http.req.version.major + "." + http.req.version.minor.sub(1)+
"r\nHost:10.101.10.10\r\nAccept: */*\r\n\r\n"}
```

3. Verify the configurations of the HTTP callout.
   show policy httpCallout <name>

To configure an HTTP callout by using the configuration utility

1. Navigate to AppExpert > HTTP Callouts.
2. In the details pane, click Add.
3. In the Create HTTP Callout dialog box, configure the parameters of the HTTP callout. For a description of the parameter, hover the mouse cursor over the check box.
4. Click Create and then click Close.

Verifying the Configuration

Aug 30, 2013

For an HTTP callout to work correctly, all the HTTP callout parameters and the entities associated with the callout must be configured correctly. While the NetScaler appliance does not check the validity of the HTTP callout parameters, it indicates the state of the bound entities, namely the server or virtual server to which the HTTP callout is sent. The following table lists the icons and describes the conditions under which the icons are displayed.

Table 1. Icons That Indicate the States of Entities Bound to an HTTP Callout

Icon	Indicates that
	The state of the server that hosts the HTTP callout agent, or the load balancing, content switching, or cache redirection virtual server to which the HTTP callout is sent is UP.
	The state of the server that hosts the HTTP callout agent, or the load balancing, content switching, or cache redirection virtual server to which the HTTP callout is sent is OUT OF SERVICE.
	The state of the server that hosts the HTTP callout agent, or the load balancing, content switching, or cache redirection virtual server to which the HTTP callout is sent is DOWN.

For an HTTP callout to function correctly, the icon must be green at all times. If the icon is not green, check the state of the callout server or virtual server to which the HTTP callout is sent. If the HTTP callout is not working as expected even though the icon is green, check the parameters configured for the callout.

You can also verify the configuration by sending test requests that match the policy from which the HTTP callout is invoked, checking the hits counter for the policy and the HTTP callout, and verifying the responses that the NetScaler appliance sends to the client.

Note: An HTTP callout can sometimes invoke itself recursively a second time. If this happens, the hits counter is incremented by two counts for each callout that is generated by the appliance. For the hits counter to display the correct value, you must configure the HTTP callout in such a way that it does not invoke itself a second time. For more information about how you can avoid HTTP callout recursion, see "Avoiding HTTP Callout Recursion."

To view the hits counter for an HTTP callout

1. Navigate to AppExpert > HTTP Callouts.
2. In the details pane, click the HTTP callout for which you want to view the hits counter, and then view the hits in the Details area.

Invoking an HTTP Callout

Aug 30, 2013

After you configure an HTTP callout, you invoke the callout by including the SYS.HTTP_CALLOUT(<name>) expression in a default syntax policy rule. In this expression, <name> is the name of the HTTP callout that you want to invoke.

You can use default syntax expression operators with the callout expression to process the response and then perform an appropriate action. The return type of the response from the HTTP callout agent determines the set of operators that you can use on the response. If the part of the response that you want to analyze is text, you can use a text operator to analyze the response. For example, you can use the CONTAINS(<string>) operator to check whether the specified portion of the response contains a particular string, as in the following example:

```
SYS.HTTP_CALLOUT(mycallout).contains("Good IP address")
```

If you use the preceding expression in a responder policy, you can configure an appropriate responder action.

Similarly, if the part of the response that you want to evaluate is a number, you can use a numeric operator such as GT(int). If the response contains a Boolean value, you can use a Boolean operator.

Note: An HTTP callout can invoke itself recursively. HTTP callout recursion can be avoided by combining the HTTP callout expression with a default syntax expression that prevents recursion. For information about how you can avoid HTTP callout recursion, see "Avoiding HTTP Callout Recursion."

You can also cascade HTTP callouts by configuring policies that each invoke a callout after evaluating previously generated callouts. In this scenario, after one policy invokes a callout, when the NetScaler appliance is parsing the callout before sending the callout to the callout server, a second set of policies can evaluate the callout and invoke additional callouts, which can in turn be evaluated by a third set of policies, and so on. Such an implementation is described in the following example.

First, you could configure an HTTP callout called myCallout1, and then configure a responder policy, Pol1, to invoke myCallout1. Then, you could configure a second HTTP callout, myCallout2, and a responder policy, Pol2. You configure Pol2 to evaluate myCallout1 and invoke myCallout2. You bind both responder policies globally.

To avoid HTTP callout recursion, myCallout1 is configured with a unique custom HTTP header called "Request1." Pol1 is configured to avoid HTTP callout recursion by using the default syntax expression, HTTP.REQ.HEADER(\"Request1\").EQ(\"Callout Request\").NOT.

Pol2 uses the same default syntax expression, but excludes the .NOT operator so that the policy evaluates myCallout1 when the NetScaler appliance is parsing it. Note that myCallout2 identifies its own unique header called "Request2," and Pol2 includes a default syntax expression to prevent myCallout2 from invoking itself recursively.

Example

```
> add policy httpCallout myCallout1

Done

> set policy httpCallout myCallout1 -IPAddress 10.102.3.95 -port 80 -returnType TEXT -hostExpr
 "\"10.102.3.95\"" -urlStemExpr "\"/cgi-bin/check_clnt_from_database.pl\"" -headers Request1
("Callout Request") -parameters cip(CLIENT.IP.SRC) -resultExpr "HTTP.RES.BODY(100)"

Done
```

> add responder policy Pol1 "HTTP.REQ.HEADER(\"Request1\").EQ(\"Callout Request\").NOT && SYS.HTTP_CALLOUT(myCallout1).CONTAINS(\"IP Matched\")" RESET

Done

> bind responder global Pol1 100 END -type OVERRIDE

Done

> add policy httpCallout myCallout2

Done

> set policy httpCallout myCallout2 -IPAddress 10.102.3.96 -port 80 -returnType TEXT -hostExpr "\"10.102.3.96\"" -urlStemExpr "\"/cgi-bin/check_clnt_location_from_database.pl\"" -headers Request2 ("Callout Request") -parameters cip(CLIENT.IP.SRC) -resultExpr "HTTP.RES.BODY(200)"

Done

> add responder policy Pol2 "HTTP.REQ.HEADER(\"Request2\").EQ(\"Callout Request\").NOT && HTTP.REQ.HEADER(\"Request1\").EQ(\"Callout Request\") && SYS.HTTP_CALLOUT(myCallout2).CONTAINS (\"APAC\")" RESET

Done

> bind responder global Pol2 110 END -type OVERRIDE

Done

Avoiding HTTP Callout Recursion

Apr 03, 2013

Even though the NetScaler appliance does not check for the validity of the HTTP callout request, it parses the request once before it sends the request to the HTTP callout agent. This parsing allows the appliance to treat the callout request as any other incoming request, which in turn allows you to configure several useful NetScaler features (such as integrated caching, SureConnect, and Priority Queuing) to work on the callout request.

However, during this parsing, the HTTP callout request can hit the same policy and therefore invoke itself recursively. The appliance detects the recursive invocation and raises an undefined (UNDEF) condition. However, the recursive invocation results in the policy and HTTP callout hit counters being incremented by two counts each instead of one count each.

To prevent a callout from invoking itself, you must identify at least one unique characteristic of the HTTP callout request, and then exclude all requests with this characteristic from being processed by the policy rule that invokes the callout. You can do so by including another default syntax expression in the policy rule. The expression must precede the SYS.HTTP_CALLOUT(<name>) expression so that it is evaluated before the callout expression is evaluated. For example:

<Expression that prevents callout recursion> && SYS.HTTP_CALLOUT(<name>)

When you configure a policy rule in this way, when the appliance generates the request and parses it, the compound rule evaluates to FALSE, the callout is not generated a second time, and the hit counters are incremented correctly.

One way by which you can assign a unique characteristic to an HTTP callout request is to include a unique custom HTTP header when you configure the callout. Following is an example of an HTTP callout called "myCallout." The callout generates an HTTP request that checks whether a client's IP address is present in a database of blacklisted IP addresses. The callout includes a custom header called "Request," which is set to the value "Callout Request." A globally bound responder policy, "Pol1," invokes the HTTP callout but excludes all requests whose Request header is set to this value, thus preventing a second invocation of myCallout. The expression that prevents a second invocation is HTTP.REQ.HEADER(\"Request\").EQ(\"Callout Request\").NOT.

Example

```
> add policy httpCallout myCallout
 Done

> set policy httpCallout myCallout -IPAddress 10.102.3.95 -port 80 -returnType TEXT -hostExpr "\"10.102.3.95\"" -urlStemExpr "\"/cgi-bin/check_clnt_from_database.pl\"" -h
Done

> add responder policy Pol1 "HTTP.REQ.HEADER(\"Request\").EQ(\"Callout Request\").NOT && SYS.HTTP_CALLOUT(myCallout).CONTAINS(\"IP Matched\")" RESET
Done

> bind responder global Pol1 100 END -type OVERRIDE
Done
```

Note: You can also configure an expression to check whether the URL of the request includes the URL stem expression that is configured for the HTTP callout. If you want to implement this scenario, make sure that the HTTP callout agent is dedicated to respond only to HTTP callouts and not to other client requests directed through the appliance. If the HTTP callout agent is an application or Web server that serves other client requests, such an expression will prevent the appliance from processing those client requests. Instead, use a unique custom header as described earlier.

Caching HTTP Callout Responses

Apr 03, 2013

For improved performance while using callouts, you can use the integrated caching feature to cache callout responses. The responses are stored in an integrated caching content group named calloutContentGroup for a specified time duration.

Note: To cache callout responses, make sure that the integrated caching feature is enabled.

To set the cache duration by using the command line interface

At the command prompt, type:

set policy httpCallout <name> -cacheForSecs <secs>

Example

```
> set httpcallout httpcallout1 -cacheForSecs 120
```

To set the cache duration by using the configuration utility

1. Navigate to AppExpert > HTTP Callouts.
2. In the details pane, select the HTTP callout for which you want to set the cache duration and click Open.
3. In the Configure HTTP Callout dialog box, specify the Cache Expiration Time.
4. Verify that you have entered the correct time duration, and then click OK.

Use Case: Filtering Clients by Using an IP Blacklist

May 21, 2015

HTTP callouts can be used to block requests from clients that are blacklisted by the administrator. The list of clients can be a publicly known blacklist, a blacklist that you maintain for you organization, or a combination of both.

The NetScaler appliance checks the IP address of the client against the pre-configured blacklist and blocks the transaction if the IP address has been blacklisted. If the IP address is not in the list, the appliance processes the transaction.

To implement this configuration, you must perform the following tasks:

1. Enable responder on the NetScaler appliance.
2. Create an HTTP callout on the NetScaler appliance and configure it with details about the external server and other required parameters.
3. Configure a responder policy to analyze the response to the HTTP callout, and then bind the policy globally.
4. Create an HTTP callout agent on the remote server.

Enabling Responder

Updated: 2013-08-30

You must enable responder before you can use it.

To enable responder by using the configuration utility

1. Make sure that you have installed the responder license.
2. In the configuration utility, expand AppExpert, and right-click Responder, and then click Enable Responder feature.

Creating an HTTP Callout on the NetScaler Appliance

Updated: 2013-08-30

Create an HTTP callout, HTTP-Callout-1, with the parameter settings shown in the following table. For more information about creating an HTTP callout, see "Configuring an HTTP Callout."

Table 1. Parameters and Values for HTTP-Callout-1

Parameter	Value
Name	HTTP-Callout-1
Server to receive callout request	
IP Address	10.103.9.95
Port	80
Request to send to the server	
Method	GET
Host Expression	10.102.3.95
URL Stem Expression	"/cgi-bin/check_clnt_from_database.pl"
Headers	
Name	Request
Value-expression	Callout Request

Parameters Parameter	Value
Name	Cip
Value-expression	CLIENT.IP.SRC
Server Response	
Return Type	TEXT
Expression to extract data from the response	HTTP.RES.BODY(100)

Configuring a Responder Policy and Binding it Globally

Updated: 2013-08-30

After you configure the HTTP callout, verify the callout configuration, and then configure a responder policy to invoke the callout. While you can create a responder policy in the Policies sub-node and then bind it globally by using the Responder Policy Manager, this demonstration uses the Responder Policy Manager to create the responder policy and bind the policy globally.

To create a responder policy and bind it globally by using the configuration utility

1. Navigate to AppExpert > Responder.
2. In the details pane, under Policy Manager, click Policy Manager.
3. In the Responder Policy Manager dialog box, click Override Global.
4. Click Insert Policy, and then, under Policy Name, click New Policy.
5. In the Create Responder Policy dialog box, do the following:
   1. In Name, type Policy-Responder-1.
   2. In Action, select RESET.
   3. In Undefined-Result Action, select Global undefined-result action.
   4. In Expression, type the following default syntax expression:
      "HTTP.REQ.HEADER(\"Request\").EQ(\"Callout Request\").NOT && SYS.HTTP_CALLOUT(HTTP-Callout-1).CONTAINS(\"IP Matched\")"
   5. Click Create, and then click Close.
6. Click Apply Changes, and then click Close.

Creating an HTTP Callout Agent on the Remote Server

You must now create an HTTP callout agent on the remote callout server that will receive callout requests from the NetScaler appliance and respond appropriately. The HTTP callout agent is a script that is different for each deployment and must be written with the server specifications in mind, such as the type of database and the scripting language supported.

Following is a sample callout agent that verifies whether the given IP address is part of an IP blacklist. The agent has been written in the Perl scripting language and uses a MYSQL database.

The following CGI script checks for a given IP address on the callout server.

```
#!/usr/bin/perl -w
print "Content-type: text/html\n\n";
      use DBI();
      use CGI qw(:standard);
#Take the Client IP address from the request query
      my $ip_to_check = param('cip');
# Where a MYSQL database is running
      my $dsn = 'DBI:mysql:BAD_CLIENT:localhost';
# Database username to connect with
      my $db_user_name = ‘dbuser’;
# Database password to connect with
      my $db_password = 'dbpassword';
      my ($id, $password);
# Connecting to the database
      my $dbh = DBI->connect($dsn, $db_user_name, $db_password);
```

```
      my $sth = $dbh->prepare(qq{ select * from bad_clnt });
      $sth->execute();
      while (my ($ip_in_database) = $sth->fetchrow_array()) {
      chomp($ip_in_database);
# Check for IP match
      if ($ip_in_database eq $ip_to_check) {
           print "\n IP Matched\n";
           $sth->finish();
           exit;
       }
      }
      print "\n IP Failed\n";
      $sth->finish();
      exit;
```

Use Case: ESI Support for Fetching and Updating Content Dynamically

May 21, 2015

Edge Side Includes (ESI) is a markup language for edge-level dynamic Web content assembly. It helps in accelerating dynamic Web-based applications by defining a simple markup language to describe cacheable and non-cacheable Web page components that can be aggregated, assembled, and delivered at the network edge. By using HTTP callouts on the NetScaler appliance, you can read through the ESI constructs and aggregate or assemble content dynamically.

To implement this configuration, you must perform the following tasks:

1. Enable rewrite on the NetScaler appliance.
2. Create an HTTP callout on the appliance and configure it with details about the external server and other required parameters.
3. Configure a rewrite action to replace the ESI content with the callout response body.
4. Configure a rewrite policy to specify the conditions under which the action is performed, and then bind the rewrite policy globally.

Enabling Rewrite

Updated: 2013-08-30

Rewrite must be enabled before it is used on the NetScaler appliance. The following procedure describes the steps to enable the rewrite feature.

To enable rewrite by using the configuration utility

1. Make sure that you have installed the rewrite license.
2. In the configuration utility, expand AppExpert, and right-click Rewrite, and then click Enable Rewrite feature.

Creating an HTTP Callout on the NetScaler Appliance

Updated: 2013-08-30

Create an HTTP callout, HTTP-Callout-2, with the parameter settings shown in the following table. For more information about creating an HTTP callout, see "Configuring an HTTP Callout."

Table 1. Parameters and Values for HTTP-Callout-2

Parameter	Value
Name	HTTP-Callout-2
Server to receive callout request	
IP Address	10.102.56.51
Port	80

Parameter	Value
Request to send to the server	
Method	GET
Host Expression	10.102.56.51:80
URL Stem Expression	"HTTP.RES.BODY(500).AFTER_STR(\"src=\").BEFORE_STR(\"/>\")"
Headers	
Name	Name
Value-expression	Callout
Server Response	
Return Type	TEXT
Expression to extract data from the response	HTTP.RES.BODY(100)

Configuring the Rewrite Action

Updated: 2013-08-30

Create a rewrite action, Action-Rewrite-1, to replace the ESI content with the callout response body. Use the parameter settings shown in the following table.

Table 2. Parameters and Values for Action-Rewrite-1

Parameter	Value
Name	Action-Rewrite-1
Type	Replace
Expression to choose target text reference	"HTTP.RES.BODY(500).AFTER_STR (\" <example>\").BEFORE_STR (\" </example>\")"
String expression for replacement text	"SYS.HTTP_CALLOUT(HTTP-Callout-2)"

To configure the rewrite action by using the configuration utility

1. Navigate to AppExpert > Rewrite > Actions.
2. In the details pane, click Add.
3. In the Create Rewrite Action dialog box, in Name, type Action-Rewrite-1.
4. In Type, select REPLACE.
5. In Expression to choose target text reference, type the following default syntax expression:
   "HTTP.RES.BODY(500).AFTER_STR(\"<example>\").BEFORE_STR(\"</example>\")"
6. In the String expression for replacement text, type the following string expression:
   "SYS.HTTP_CALLOUT(HTTP-Callout-2)"
7. Click Create, and then click Close.

Creating the Rewrite Policy and Binding it Globally

Updated: 2013-08-30

Create a rewrite policy, Policy-Rewrite-1, with the parameter settings shown in the following table. You can create a rewrite policy in the Policies subnode and then bind it globally by using the Rewrite Policy Manager. Alternatively, you can use the Rewrite Policy Manager to perform both these tasks simultaneously. This demonstration uses the Rewrite Policy Manager to perform both tasks.

Table 3. Parameters and Values for Policy-Rewrite-1

Parameter	Value
Name	Policy-Rewrite-1
Action	Action_Rewrite-1
Undefined Result Action	-Global undefined-result action-
Expression	"HTTP.REQ.HEADER(\"Name\").CONTAINS (\"Callout\").NOT"

To configure a rewrite policy and bind it globally by using the configuration utility

1. Navigate to AppExpert > Rewrite.
2. In the details pane, under Policy Manager, click Rewrite Policy Manager.
3. In the Rewrite Policy Manager dialog box, click Override Global.
4. Click Insert Policy, and then, in the Policy Name column, click New Policy.
5. In the Create Rewrite Policy dialog box, do the following:
   1. In Name, type Policy-Rewrite-1.
   2. In Action, select Action-Rewrite-1.
   3. In Undefined-Result Action, select Global undefined-result action.
   4. In Expression, type the following default syntax expression:
      "HTTP.REQ.HEADER(\"Name\").CONTAINS(\"Callout\").NOT"
   5. Click Create, and then click Close.
6. Click Apply Changes, and then click Close.

Use Case: Access Control and Authentication

May 21, 2015

In high security zones, it is mandatory to externally authenticate the user before a resource is accessed by clients. On the NetScaler appliance, you can use HTTP callouts to externally authenticate the user by evaluating the credentials supplied. In this example, the assumption is that the client is sending the user name and password through HTTP headers in the request. However, the same information could be fetched from the URL or the HTTP body.

To implement this configuration, you must perform the following tasks:

1. Enable the responder feature on the NetScaler appliance.
2. Create an HTTP callout on the appliance and configure it with details about the external server and other required parameters.
3. Configure a responder policy to analyze the response, and then bind the policy globally.
4. Create a callout agent on the remote server.

Enabling Responder

Updated: 2013-08-30

The responder feature must be enabled before it is used on the NetScaler appliance.

To enable responder by using the configuration utility

1. Make sure that the responder license is installed.
2. In the configuration utility, expand AppExpert, and right-click Responder, and then click Enable Responder feature.

Creating an HTTP Callout on the NetScaler Appliance

Updated: 2013-08-30

Create an HTTP callout, HTTP-Callout-3, with the parameter settings shown in the following table. For more information about creating an HTTP callout, see "Configuring an HTTP Callout."

Table 1. Parameters and Values for HTTP-Callout-3

Parameter	Value
Name	HTTP-Callout-3
Server to receive callout request	
IP Address	10.103.9.95
Port	80
Request to send to the server	
Method	GET
Host Expression	10.102.3.95
URL Stem Expression	"/cgi-bin/authenticate.pl"
Headers	
Name	Request
Value-expression	Callout Request
Parameters	
Name	Username
Value-expression	HTTP.REQ.HEADER("Username").VALUE(0)
Name	Password
Value-expression	HTTP.REQ.HEADER("Password").VALUE(0)

Parameter Server Response	Value
Return Type	TEXT
Expression to extract data from the response	HTTP.RES.BODY(100)

Creating a Responder Policy to Analyze the Response

Updated: 2013-08-30

Create a responder policy, Policy-Responder-3, that will check the response from the callout server and RESET the connection if the source IP address has been blacklisted. Create the policy with the parameters settings shown in the following table. While you can create a responder policy in the Policies subnode and then bind it globally by using the Responder Policy Manager, this demonstration uses the Responder Policy Manager to create the responder policy and bind the policy globally.

Table 2. Parameters and Values for Policy-Responder-3

Parameter	Value
Name	Policy-Responder-3
Action	RESET
Undefined-Result-Action	-Global undefined-result action-
Expression	"HTTP.REQ.HEADER(\"Request\").EQ(\"Callout Request\").NOT && SYS.HTTP_CALLOUT(HTTP-Callout-3).CONTAINS(\"Authentication Failed\")"

To create a responder policy and bind it globally by using the configuration utility

1. Navigate to AppExpert > Responder.
2. In the details pane, under Policy Manager, click Responder Policy Manager.
3. In the Responder Policy Manger dialog box, click Override Global.
4. Click Insert Policy, and then, in the Policy Name column, click New Policy.
5. In the Create Responder Policy dialog box, do the following:
   1. In Name, type Policy-Responder-3.
   2. In Action, select RESET.
   3. In Undefined-Result Action , select Global undefined-result action.
   4. In the Expression text box, type:
      "HTTP.REQ.HEADER(\"Request\").EQ(\"Callout Request\").NOT && SYS.HTTP_CALLOUT(HTTP-Callout-3).CONTAINS(\"Authentication Failed\")"
   5. Click Create, and then click Close.
6. Click Apply Changes, and then click Close.

Creating an HTTP Callout Agent on the Remote Server

You now need to create an HTTP callout agent on the remote callout server. The HTTP callout agent receives callout requests from the NetScaler appliance and responds appropriately. The callout agent is a script that is different for each deployment and must be written with server specifications in mind, such as the type of database and the scripting language supported.

Following is sample callout agent pseudo-code that verifies whether the supplied user name and password are valid. The agent can be implemented in any programming language of your choice. The pseudo-code is to be used only as a guideline for developing the callout agent. You can build additional functionality into the program.

To verify the supplied user name and password by using pseudo-code

1. Accept the user name and password supplied in the request and format them appropriately.
2. Connect to the database that contains all the valid user names and passwords.
3. Check the supplied credentials against your database.
4. Format the response as required by the HTTP callout.
5. Send the response to the NetScaler appliance.

Use Case: OWA-Based Spam Filtering

May 21, 2015

Spam filtering is the ability to dynamically block emails that are not from a known or trusted source or that have inappropriate content. Spam filtering requires an associated business logic that indicates that a particular kind of message is spam. When the NetScaler appliance processes Outlook Web Access (OWA) messages based on the HTTP protocol, HTTP callouts can be used to filter spam.

You can use HTTP callouts to extract any portion of the incoming message and check with an external callout server that has been configured with rules that are meant for determining whether a message is legitimate or spam. In case of spam email, for security reasons, the NetScaler appliance does not notify the sender that the email is marked as spam.

The following example conducts a very basic check for various listed keywords in the email subject. These checks can be more complex in a production environment.

To implement this configuration, you must perform the following tasks:

1. Enable the responder feature on the NetScaler appliance.
2. Create an HTTP callout on the NetScaler appliance and configure it with details about the external server and other required parameters.
3. Create a responder policy to analyze the response, and then bind the policy globally.
4. Create a callout agent on the remote server.

Enabling Responder

Updated: 2013-08-30

The responder feature must be enabled before it can be used on the NetScaler appliance.

To enable responder by using the configuration utility

1. Make sure that the responder license is installed.
2. In the configuration utility, expand AppExpert, and right-click Responder, and then click Enable Responder feature.

Creating an HTTP Callout on the NetScaler Appliance

Updated: 2013-08-30

Create an HTTP callout, HTTP-Callout-4, with the parameter settings shown in the following table. For more information about creating an HTTP callout, see "Configuring an HTTP Callout."

Table 1. Parameters and Values for HTTP-Callout-4

Parameter	Value
Name	HTTP-Callout-4
Server to receive callout request	
IP Address	10.103.56.51
Port	80
Request to send to the server	
Method	POST
Host Expression	fffffff
URL Stem Expression	"/cgi-bin/Callout/spam_filter.pl"
Headers	
Name	Request
Value-expression	Callout Request
Parameters	
Name	Subject
Value-expression	("\"" + HTTP.REQ.BODY(1000).AFTER_STR("urn:schemas:httpmail:subject=").BEFORE_STR("\n").TO_LOWER + "\"")
Server Response	

Parameter	Value
Return Type	BOOL
Expression to extract data from the response	HTTP.RES.BODY(100) .CONTAINS(\"Matched\")

Creating a Responder Action

Updated: 2013-08-30

Create a responder action, Action-Responder-4. Create the action with the parameter settings shown in the following table.

Table 2. Parameters and Values for Action-Responder-4

Parameter	Value
Name	Action-Responder-4
Type	Respond with
Target	"\"HTTP/1.1 200 OK\r\nServer: Microsoft-IIS/6.0\r\nX-Powered-By: ASP.NET\r\nContent-Length: 0\r\nMS-WebStorage: 6.5.6944\r\nCache-Control: no-cache\r\n\r\n\""

To create a responder action by using the configuration utility

1. Navigate to AppExpert > Responder > Actions.
2. In the details pane, click Add.
3. In the Create Responder Action dialog box, in Name, type Action-Responder-4.
4. In Type, click Respond with.
5. In Target, type:
   "\"HTTP/1.1 200 OK\r\nServer: Microsoft-IIS/6.0\r\nX-Powered-By: ASP.NET\r\nContent-Length: 0\r\nMS-WebStorage: 6.5.6944\r\nCache-Control: no-cache\r\n\r\n\""
6. Click Create, and then click Close.

Creating a Responder Policy to Invoke the HTTP Callout

Updated: 2013-08-30

Create a responder policy, Policy-Responder-4, that will check the request body and, if the body contains the word "*subject*," invoke the HTTP callout to verify the email. Create the policy with the parameter settings shown in the following table. While you can create a responder policy in the Policies subnode and then bind it globally by using the Responder Policy Manager, this demonstration uses the Responder Policy Manager to create the responder policy and bind it globally.

Table 3. Parameters and Values for Policy-Responder-4

Parameter	Value
Name	Policy-Responder-4
Action	Action-Responder-4
Undefined-Result-Action	-Global undefined-result action-
Expression	"HTTP.REQ.BODY(1000).CONTAINS(\"urn:schemas:httpmail:subject\") && SYS.HTTP_CALLOUT(HTTP-Callout-4)"

To create a responder policy by using the configuration utility

1. Navigate to AppExpert > Responder.
2. In the details pane, under Policy Manager, click Responder policy manager.
3. In the Responder Policy Manger dialog box, click Override Global.
4. Click Insert Policy, and then, in the Policy Name column, click New Policy.
5. In the Create Responder Policy dialog box, do the following:
   1. In Name, type Policy-Responder-4.
   2. In Action, click Action-Responder-4.
   3. In Undefined-Result Action, click Global undefined-result action.
   4. In the Expression text box, type:
      "HTTP.REQ.BODY(1000).CONTAINS(\"urn:schemas:httpmail:subject\") && SYS.HTTP_CALLOUT(HTTP-Callout-4)"
   5. Click Create, and then click Close.
6. Click Apply Changes, and then click Close.

Creating an HTTP Callout Agent on the Remote Server

You will now need to create an HTTP callout agent on the remote callout server. The HTTP callout agent receives callout requests from the NetScaler appliance and responds accordingly. The callout agent is a script that is different for each deployment and must be written with server specifications in mind, such as the type of database and the scripting language supported.

The following pseudo-code provides instructions for creating a callout agent that checks a list of words that are generally understood to indicate spam mails. The agent can be implemented in any programming language of your choice. The pseudo-code is to be used only as a guideline for developing the callout agent. You can build additional functionality into the program.

To identify spam email by using pseudo-code

1. Accept the email subject provided by the NetScaler appliance.
2. Connect to the database that contains all the terms against which the email subject is checked.
3. Check the words in the email subject against the spam word list.
4. Format the response as required by the HTTP callout.
5. Send the response to the NetScaler appliance.

Use Case: Dynamic Content Switching

May 21, 2015

This use case provides dynamic content switching by using an HTTP callout to get the name of the load balancing virtual server to which the request is forwarded.

1. Add a content switching virtual server.
   > add cs vserver cs_vserver1 HTTP 10.102.29.196 80
2. Create an HTTP callout.
   > add policy httpCallout http_callout1
3. Configure the HTTP callout to respond with the name of the load balancing virtual server from a request that contains the client IP address in the HTTP header "X-CLIENT-IP".
   > set policy httpCallout http_callout1 -IPAddress 10.217.14.23 -port 80 -returnType TEXT -hostExpr "\"www.get-lbvip.com\"" -urlStemExpr "\"/index.html\"" -headers X-CL
4. Configure the content switching action to retrieve the callout response.
   > add cs action cs_action1 -targetVserverExpr 'SYS.HTTP_CALLOUT(http_callout1)'
   Note: You must bind a load balancing virtual server to the content switching virtual server to account for:
   - The non-availability of the load balancing virtual server that the callout resolves to.
   - A UNDEF condition that results from the execution of the callout.

   > bind cs vserver cs_vserver1 -lbvserver default_lbvip
5. Configure the content switching policy.
   > add cs policy cs_policy1 -rule true -action cs_action1
6. Binding the content switching policy to the content switching virtual server.
   > bind cs vserver cs_vserver1 -policyName cs_policy1 -priority 10

Pattern Sets and Data Sets

Jun 20, 2013

Policy expressions for string matching operations on a large set of string patterns tend to become long and complex. Resources consumed by the evaluation of such complex expressions are significant in terms of processing cycles, memory, and configuration size. You can create simpler, less resource-intensive expressions by using pattern matching.

Depending on the type of patterns that you want to match, you can use one of the following features to implement pattern matching:

- A pattern set is an array of indexed patterns used for string matching during default syntax policy evaluation. Example of a pattern set: imagetypes {svg, bmp, png, gif, tiff, jpg}.
- A data set is a specialized form of pattern set. It is an array of patterns of types number (integer), IPv4 address, or IPv6 address.

In many cases, you can use either pattern sets or data sets. However, in cases where you want specific matches for numerical data or IPv4 and IPv6 addresses, you must use data sets.

Note: Pattern sets and data sets can be used only in default syntax policies.
To use pattern sets or data sets, first create the pattern set or data set and bind patterns to it. Then, when you configure a policy for comparing a string in a packet, use an appropriate operator and pass the name of the pattern set or data set as an argument.

How String Matching works with Pattern Sets and Data Sets

Jul 11, 2013

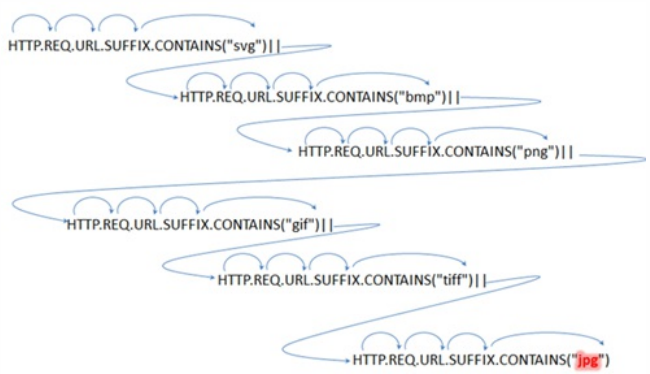
A pattern set or data set contains a set of patterns, and each pattern is assigned a unique index. When a policy is applied to a packet, an expression identifies a string to be evaluated, and the operator compares the string to the patterns defined in the pattern set or data set until a match is found or all patterns have been compared. Then, depending on its function, the operator returns either a boolean value that indicates whether or not a matching pattern was found or the index of the pattern that matches the string.

Note: This topic explains the working of a pattern set. Data sets work the same way. The only difference between pattern sets and data sets is the type of patterns defined in the set.
Consider the following use case to understand how patterns can be used for string matching.

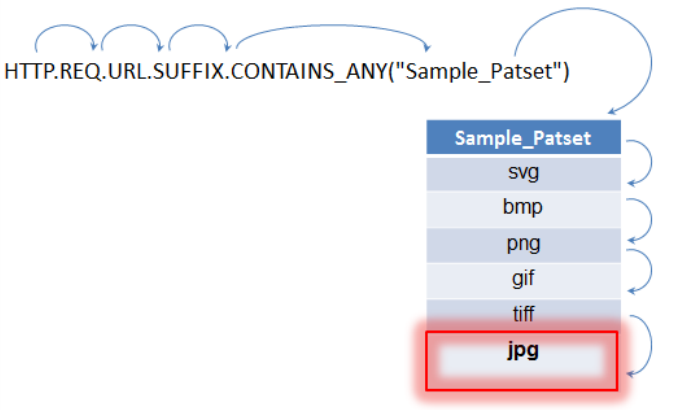
You want to determine whether the URL suffix (target text) contains any of the image file extensions. Without using pattern sets, you would have to define a complex expression, as follows:

```
HTTP.REQ.URL.SUFFIX.CONTAINS("svg") ¦¦ HTTP.REQ.URL.SUFFIX.CONTAINS("bmp") ¦¦ HTTP.REQ.URL.SUFFIX.CONTAINS("png") ¦¦ HTTP.REQ.URL.SUFFIX.CONTAINS("gif") ¦¦ HTTP.REQ.URL.SUFFIX.CONTAINS("tiff") ¦¦ HTTP.REQ.URL.SUFFIX.CONTAINS("jpg")
```

If the URL has a suffix of "jpg," with the above compound expression, the NetScaler appliance has to iterate through the entire compound expression sequentially, from one sub expression to the next, to determine that the request refers to a jpg image. The following figure shows the steps in the process.



When a compound expression includes hundreds of sub expressions, the above process is resource intensive. A better alternative is an expression that invokes a pattern set, as shown in the following figure.



During policy evaluation as shown above, the operator (CONTAINS_ANY) compares the string identified in the request with the patterns defined in the pattern set until a match is found. With the `Sample_Patset` expression, the multiple iterations through six sub expressions are reduced to just one.

By eliminating the need to configure compound expressions that perform string matching with multiple OR operations, pattern sets or data sets simplify configuration and accelerate processing of requests and responses.

Configuring a Pattern Set

Oct 29, 2013

To configure a pattern set, you must specify the strings that are to serve as patterns. You can manually assign a unique index value to each of these patterns, or you can allow the index values to be assigned automatically.

Note: Pattern sets are case sensitive (unless you specify the expression to ignore case). Therefore, the string pattern "product1," for example, is not the same as the string pattern "Product1."

Points to remember about index values

- You cannot bind the same index value to more than one pattern.
- An automatically assigned index value is one number larger than the highest index value of the existing patterns within the pattern set. For example, if the highest index value of existing patterns in a pattern set is 104, the next automatically assigned index value will be 105.
- If you do not specify an index for the first pattern, index value 1 is automatically assigned to that pattern.
- Index values are not regenerated automatically if one or more patterns are deleted or modified. For example, if the set contains five patterns, with indexes from 1 through 5, and if the pattern with an index of 3 is deleted, the other index values in the pattern set are not automatically regenerated to produce values from 1 through 4.
- The maximum index value that can be assigned to a pattern is 4294967290. If that value is already assigned to a pattern in the set, you must manually assign index values to any newly added patterns. An unused index value that is lower than a currently used value cannot be assigned automatically.

At the command prompt, do the following:

1. Create a pattern set.
   add policy patset <name>

   Example:
   ```
   > add policy patset samplepatset
   ```
2. Bind patterns to the pattern set.
   bind policy patset <name> <string> [-index <positive_integer>]

   Example:
   ```
   > bind policy patset samplepatset product1 -index 1
   ```
   Note: Repeat this step for all the patterns you want to bind to the pattern set.
3. Verify the configuration.
   show policy patset <name>

1. Navigate to AppExpert > Pattern Sets.
2. In the details pane, click Add to open the Create Pattern Set dialog box.
3. Specify a name for the pattern set in the Name text box.
4. Under Specify Pattern, type the first pattern and, optionally, specify values for the following parameters:
   - Treat back slash as escape character—Select this check box to specify that any backslash characters that you might include in the pattern are to be treated as escape characters.
   - Index—A user assigned index value, from 1 through 4294967290.

5. Verify that you have entered the correct characters, and then click Add.
6. Repeat steps 4 and 5 to add additional patterns, and then click Create.

Configuring a Data Set

Jul 30, 2014

To configure a data set, you must specify the strings that are to serve as patterns, and assign a type (number, IPv4 address, or IPv6 address) to each pattern. You can manually assign a unique index value to each of these patterns, or you can allow the index values to be assigned automatically.

Note: Data sets are case sensitive (unless you specify the expression to ignore case). Therefore, the string pattern "product1," for example, is not the same as the string pattern "Product1."

The rules applied for index values of data sets are the same as those applied for pattern sets. For information about index values, see "Configuring a Pattern Set."

At the command prompt, do the following:

1. Create a data set.
   add policy dataset <name> <type>

   Example:

   ```
   > add policy dataset sampledataset ipv4
   ```
2. Bind patterns to the data set.
   bind policy dataset <name> <value> [-index <positive_integer>]

   Example:

   ```
   > bind policy dataset sampledataset 10.102.29.1 -index 1
   ```
   Note: Repeat this step for all the patterns you want to bind to the data set.
3. Verify the configuration.
   show policy dataset <name>

Navigate to AppExpert > Data Sets, click Add and specify the relevant details.

Using Pattern Sets and Data Sets

Aug 30, 2013

Default syntax policy expressions that take pattern sets or data sets as an argument can be used to perform string matching operations.

The usage is as follows:
<text>.<operator>("<name>")
where,

- <text> is the expression that identifies a string in a packet. Example: HTTP.REQ.HEADER("Host").
- <operator> is one of the operators described in the following table.

Table 1. Operators for pattern sets and data sets

Operator	Description
<text>.CONTAINS_ANY(<name>)	Returns true if the target text contains one or more of the patterns defined in the specified pattern set or data set.
<text>.SUBSTR_ANY(<name>)	Returns the first string that matches any pattern defined in the specified pattern set or data set.
<text>.BEFORE_STR_ANY(<name>)	Returns the text that is present before the first occurrence of any of the patterns defined in the specified pattern set or data set.
<text>.AFTER_STR_ANY(<name>)	Returns the text that is present after the first occurrence of any of the patterns defined in the specified pattern set or data set.
<text>.EQUALS_ANY (<name>)	Returns true if the target text exactly matches any of the patterns defined in the specified pattern set or data set.
<text>.ENDSWITH_ANY(<name>)	Returns true if the target text ends with any of the patterns that are defined in the specified pattern set or data set.
<text>.STARTSWITH_ANY(<name>)	Returns true if the target text starts with any of the patterns that are defined in the specified pattern set or data set.
<text>.STARTSWITH_INDEX(<name>)	Evaluates whether the target text starts with any of the patterns that are defined in the specified pattern set or data set. If a match is found, the index of the matching pattern is returned. Otherwise, 0 is returned.
<text>.ENDSWITH_INDEX(<name>)	Evaluates whether the target text ends with any of the patterns that are defined in the specified pattern set or data set. If a match is found, the index of the matching pattern is returned. Otherwise, 0 is returned.
<text>.CONTAINS_INDEX(<name>)	Evaluates whether the target text contains any of the patterns that

Operator	Description
	are defined in the specified pattern set or data set. If a match is found, the index of the matching pattern is returned. Otherwise, 0 is returned.
<text>.EQUALS_INDEX(<name>)	Evaluates whether the target text exactly matches any of the patterns that are defined in the specified pattern set or data set. If an exact match is found, the index of the pattern is returned. Otherwise, 0 is returned.

- <name> is the name of the pattern set or data set

For sample usage, see "Sample Usage."

Sample Usage

Jun 13, 2013

To understand the usage of pattern sets in expressions, consider the example of a pattern set named "imagetypes."

Table 1. Pattern set "imagetypes"

Patterns	Index value
svg	1
bmp	2
png	3
gif	4
tiff	5
jpg	6

Example 1: Determine whether the suffix of an HTTP request is one of the file extensions defined in the "imagetypes" pattern set.

- **Expression.** HTTP.REQ.URL.SUFFIX.EQUALS_ANY("imagetypes")
- **Sample URL.** http://www.example.com/homepageicon.jpg
- **Result.** TRUE

Example 2: Determine whether the suffix of an HTTP request is one of the file extensions defined in the "imagetypes" pattern set, and return the index of that pattern.

- **Expression.** HTTP.REQ.URL.SUFFIX.EQUALS_INDEX("imagetypes")
- **Sample URL.** http://www.example.com/mylogo.gif
- **Result.** 4 (The index value of the pattern "gif".)

Example 3: Use the index value of a pattern to determine whether the URL suffix is within a specified index-value range.

- **Expression.** HTTP.REQ.URL.SUFFIX.EQUALS_INDEX("imagetypes").GE(3) && HTTP.REQ.URL.SUFFIX.EQUALS_INDEX("imagetypes").LE(5)
- **Sample URL.** http://www.example.com/mylogo.gif
- **Result.** TRUE (The index value of gif file types is 4.)

Example 4: Implement one set of policies for file extensions bmp, jpg, and png, and a different set of policies for gif, tiff, and svg files.

An expression that returns the index of a matched pattern can be used to define traffic subsets for a web application. The

following two expressions could be used in content switching policies for a content switching virtual server:

- HTTP.REQ.URL.SUFFIX.EQUALS_INDEX("imagetypes").LE(3)
- HTTP.REQ.URL.SUFFIX.EQUALS_INDEX("imagetypes").GE(4)

Variables

Jun 02, 2015

Note: Supported only on NetScaler release 10.1.e.

Variables are named objects that store information in the form of tokens. These tokens are used within and across different transactions on the NetScaler Appliance for internal computation and policy processing.

The NetScaler appliance supports creation of variables of the following types:

- **Singleton variables.** Can have a single value of one of the following types: ulong and text (max-size). The ulong type is an unsigned 64-bit integer, the text type is a sequence of bytes, and max-size is the maximum number of bytes in the sequence.
- **Map variables.** Maps hold values associated with keys: each key-value pair is called a map entry. The key for each entry is unique within the map. Maps are specified as follows:

  map (key_type, value_type, max-values).

  where,

  - *key_type* is the data type of the key. It is of type text (max-size).
  - *value_type* is the data type of the values of the map. It can be of type ulong or text (max-size).
  - *max-values* is the maximum number of entries that the map can contain. It is of type ulong.

Values for these variables are set using assignments which must be invoked on policy actions.

Note: Variables are not yet supported in a high-availability setup or in a cluster.

A map variable or a singleton variable can have a global scope. Alternatively, the scope of a singleton variable can be limited to a single transaction.

- **Global Scope Variable** - A variable with global scope (the default) has only one instance, and that instance has the same value(s) across all cores of a NetScaler appliance and across all nodes of a cluster or HA configuration. Global variable values exist until they are explicitly deleted, until they expire, or until a standalone appliance is restarted or all nodes of a cluster or HA configuration are restarted.
- **Transaction Scope Variable** - A variable with transaction scope has a separate instance, with its own value, for each transaction processed by the NetScaler appliance. When the transaction processing is complete, the transaction variable value is deleted.

Note: Transaction scope variables are available in NetScaler release 10.5.e or later.

Configuring and Using Variables

Mar 20, 2014

You must first create a variable and then assign a value or specify the operation that must be performed on the variable. After performing these operations, you can use the assignment as a policy action.

Note: Once configured, a variable's settings cannot be modified or reset. If the variable needs to be changed, the variable and all references to the variable (expressions and assignments) must be deleted. The variable can then be re-added with new settings, and the references (expressions and assignments) can be re-added.

1. Create a variable.
   add ns variable <name> -type <string> [-scope global] [-ifFull (undef | lru)] [-ifValueTooBig (undef | truncate)] [-ifNoValue (undef | init)] [-init <string>] [-expires <positive_integer>] [-comment <string>]

   Note: Refer to the man page "man add ns variable" for description of the command parameters.
   Example 1: Create a ulong variable named "my_counter" and initialize it to 1.
   `add ns variable my_counter –type ulong -init 1`
   Example 2: Create a map named "user_privilege_map". The map will contain keys of maximum length 15 characters and text values of maximum length 10 characters, with a maximum of 10000 entries.
   `add ns variable user_privilege_map -type map(text(15),text(10),10000)`
   Note: If the map contains 10000 unexpired entries, assignments for new keys reuse one of the least recently used entries. By default, an expression trying to get a value for a non-existent key will initialize an empty text value.
2. Assign the value or specify the operation to be performed on the variable. This is done by creating an assignment.
   add ns assignment <name> -variable <expression> [-set <expression> | -add <expression> | -sub <expression> | -append <expression> | -clear] [-comment <string>]

   Note: A variable is referenced by using the variable selector ($). Therefore, **$variable1** is used to refer to text or ulong variables. Similarly, **$variable2[key-expression]** is used to refer to map variables.
   Example 1: Define an assignment named "inc_my_counter" that automatically adds 1 to the "my_counter" variable.
   `add ns assignment inc_my_counter -variable $my_counter -add 1`
   Example 2: Define an assignment named "set_user_privilege" that adds to the "user_privilege_map" variable an entry for the client's IP address with the value returned by the "get_user_privilege" HTTP callout.
   `add ns assignment set_user_privilege -variable $user_privilege_map[client.ip.src.typecast_text_t] -set sys.http.callout(get_user_privilege)`
   Note: If an entry for that key already exists, the value will be replaced. Otherwise a new entry for the key and value will be added. Based on the previous declaration for user_privilege_map, if the map already has 10000 entries, one of the least recently used entries will be reused for the new key and value.
3. Invoke the variable assignment in a policy.
   There are two functions that can operate on map variables.

   - **$name.valueExists(key-expression).** Returns true if there is a value in the map selected by the key-expression. Otherwise returns false. This function will update the expiration and LRU information if the map entry exists, but will not create a new map entry if the value does not exist.
   - **$name.valueCount.** Returns the number of values currently held by the variable. This is the number of entries in a map. For a singleton variable, this is 0 if the variable is uninitialized or 1 otherwise.

   Example: Invoke the assignment named "set_user_privilege" with a compression policy.
   `> add cmp policy set_user_privilege_pol -rule $user_privilege_map.valueExists(client.ip.src.typecast_text_t).not -resAction set_user_privilege`

1. Navigate to AppExpert > NS Variables, to create a variable.
2. Navigate to AppExpert > NS Assignments, to assign value(s) to the variable.
3. Navigate to the appropriate feature area where you want to configure the assignment as an action.

Use Case: Caching User Privileges

Aug 13, 2014

In this use case, user privileges ("GOLD", "SILVER", and so on) must be retrieved from an external web service.

To achieve this use case, perform the following operations:

1. Create an HTTP callout to fetch the user privileges from the external web service.
   ```
   > add policy httpcallout get_user_privilege
   > set policy httpcallout get_user_privilege -ipaddress 10.217.193.84 -port 80 -returntype text -httpmethod get -urlstemexpr '"/get_user_privilege"' -resultexpr 'http.res.b
   ```
2. Store the privileges in a variable.
   ```
   > add ns variable user_privilege_map -type map(text(15),text(10),10000) -expires 1200
   > add ns assignment set_user_privilege -variable $user_privilege_map[client.ip.src] -set sys.http_callout(get_user_privilege)
   ```
3. Create a policy to check if there is already a cached entry for the client's IP address; if not, it calls the HTTP callout to set a map entry for the client.
   ```
   > add cmp policy set_user_privilege_pol -rule $user_privilege_map.valueExists(client.ip.src).not -resAction set_user_privilege
   ```
4. Create a policy that compresses if the cached privilege entry for the client is "GOLD".
   ```
   > add cmp policy compress_if_gold_privilege_pol -rule '$user_privilege_map[client.ip.src].eq("GOLD")' -resAction compress
   ```
5. Bind the compression policies globally.
   ```
   > bind cmp global set_user_privilege_pol -priority 10 NEXT
   > bind cmp global compress_if_gold_privilege_pol -priority 20 END
   ```

Use Case: Limiting the Number of Sessions

Jun 17, 2015

In this use case, the requirement is to limit the number of active backend sessions. In the deployment, each session login has login in the URL and each session logout has logout in the URL. On successful login, the backend sets a sessionid cookie with a unique 10 character value.

To achieve this use case, perform the following operations:

1. Create a map variable that can store each active session. The key of the map is the sessionid. The expiry time for the variable is set to 600 seconds (10 minutes).
   ```
   > add ns variable session_map -type map(text(10),ulong,100) -expires 600
   ```
2. Create the following assignments for the map variable:
   - Create an entry for the sessionid and set that value to 1 (this value is not actually used).
     ```
     > add ns assignment add_session -variable '$session_map[http.req.cookie.value("sessionid")]' -set 1
     ```
   - Deallocate the entry for a session ID, which implicitly decrements the value count for session_map.
     ```
     > add ns assignment delete_session -variable '$session_map[http.req.cookie.value("sessionid")]' -clear
     ```
3. Create responder policies for the following:
   - To check if a map entry exists for that sessionid in the HTTP request. The add_session assignment is executed if the map entry does not exist.
     ```
     > add responder policy add_session_pol 'http.req.url.contains("twbkwbis.P_SabanciLogin") || $session_map.valueExists(http.req.cookie.value("netsuis"))' add_sessior
     ```
     Note: The valueExists() function in the add_session_pol policy counts as a reference to the session's map entry, so each request resets the expiration timeout for its session. If no requests for a session are received after 10 minutes, the session's entry will be deallocated.
   - To check when the session is logged out. The delete_session assignment is executed.
     ```
     > add responder policy delete_session_pol "http.req.url.contains(\"Logout\")" delete_session
     ```
   - To check for login requests and if the number of active sessions exceed 100. If these conditions are satisfied, in order to limit the number of sessions, the user is redirected to a page that indicates that the server is busy.
     ```
     > add responder action redirect_too_busy redirect "/too_busy.html"
     > add responder policy check_login_pol "http.req.url.contains(\"twbkwbis.P_SabanciLogin\") && $session_map.valueCount > 1" redirect_too_busy
     ```
4. Bind the responder policies globally.
   ```
   > bind responder global add_session_pol 10 next
   > bind responder global delete_session_pol 10
   > bind responder global check_login_pol 20
   ```

Policies and Expressions

May 26, 2015

The following topics provide the conceptual and reference information that you require for configuring advanced policies on the Citrix® NetScaler® appliance.

You can also download a list of all the expressions supported on the NetScaler appliance and the hierarchical order in which they can be invoked. The reference is in a zip file which you can download from:

- For NetScaler 10.5: http://support.citrix.com/article/CTX141344
- For NetScaler 10.1: http://support.citrix.com/article/CTX137705

Introduction to Policies and Expressions	Describes the purpose of expressions, policies, and actions, and how different NetScaler applications make use of them.
Configuring Advanced Policies	Describes the structure of advanced policies and how to configure them individually and as policy banks.
Configuring Advanced Expressions: Getting Started	Describes expression syntax and semantics, and briefly introduces how to configure expressions and policies.
Advanced Expressions: Evaluating Text	Describes expressions that you configure when you want to operate on text (for example, the body of an HTTP POST request or the contents of a user certificate).
Advanced Expressions: Working with Dates, Times, and Numbers	Describes expressions that you configure when you want to operate on any type of numeric data (for example, the length of a URL, a client's IP address, or the date and time that an HTTP request was sent).
Advanced Expressions: Parsing HTTP, TCP, and UDP Data	Describes expressions for parsing IP and IPv6 addresses, MAC addresses, and data that is specific to HTTP and TCP traffic.
Advanced Expressions: Parsing SSL Certificates	Describes how to configure expressions for SSL traffic and client certificates, for example, how to retrieve the expiration date of a certificate or the certificate issuer.
Advanced Expressions: IP and MAC Addresses, Throughput, VLAN IDs	Describes expressions that you can use to work with any other client- or server-related data not discussed in other chapters.
Typecasting Data	Describes expressions for transforming data of one type to another.
Regular Expressions	Describes how to pass regular expressions as arguments to operators in advanced

	expressions.
Configuring Classic Policies and Expressions	Provides details on how to configure the simpler policies and expressions known as classic policies and classic expressions.
Expressions Reference	A reference for classic and advanced expression arguments.
Summary Examples of Advanced Expressions and Policies	Examples of classic and advanced expressions and policies, in both quick reference and tutorial format, that you can customize for your own use.
Tutorial Examples of Advanced Policies for Rewrite	Examples of advanced policies for use in the Rewrite feature.
Tutorial Examples of Classic Policies	Examples of classic policies for NetScaler features such as application firewall and SSL.
Migration of Apache mod_rewrite Rules to Advanced Policies	Examples of functions that were written using the Apache HTTP Server mod_rewrite engine, with examples of these functions after translation into Rewrite and Responder policies on the NetScaler.

Introduction to Policies and Expressions

Jun 04, 2015

For many NetScaler features, policies control how a feature evaluates data, which ultimately determines what the feature does with the data. A policy uses a logical expression, also called a rule, to evaluate requests, responses, or other data, and applies one or more actions determined by the outcome of the evaluation. Alternatively, a policy can apply a profile, which defines a complex action.

Some NetScaler features use default syntax policies, which provide greater capabilities than do the older, classic, policies. If you migrated to a newer release of the NetScaler software and have configured classic policies for features that now use default syntax policies, you might have to manually migrate policies to the default syntax.

This document contains the following details:

- Classic and Default Syntax Policies
- Classic and Default Syntax Expressions
- Converting Classic Expressions to the Newer Default Expression Syntax
- Before You Proceed

Classic and Default Syntax Policies

May 25, 2015

Classic policies evaluate basic characteristics of traffic and other data. For example, classic policies can identify whether an HTTP request or response contains a particular type of header or URL.

Default syntax policies can perform the same type of evaluations as classic policies. In addition, default syntax policies enable you to analyze more data (for example, the body of an HTTP request) and to configure more operations in the policy rule (for example, transforming data in the body of a request into an HTTP header).

In addition to assigning a policy an action or profile, you bind the policy to a particular point in the processing associated with the NetScaler features. The bind point is one factor that determines when the policy will be evaluated.

This document includes the following details:

Default syntax policies use a powerful expression language that is built on a class-object model, and they offer several options that enhance your ability to configure the behavior of various NetScaler features. With default syntax policies, you can do the following:

- Perform fine-grained analyses of network traffic from layers 2 through 7.
- Evaluate any part of the header or body of an HTTP or HTTPS request or response.
- Bind policies to the multiple bind points that the default syntax policy infrastructure supports at the default, override, and virtual server levels.
- Use goto expressions to transfer control to other policies and bind points, as determined by the result of expression evaluation.
- Use special tools such as pattern sets, policy labels, rate limit identifiers, and HTTP callouts, which enable you to configure policies effectively for complex use cases.

Additionally, the configuration utility extends robust graphical user interface support for default syntax policies and expressions and enables users who have limited knowledge of networking protocols to configure policies quickly and easily. The configuration utility also includes a policy evaluation feature for default syntax policies. You can use this feature to evaluate a default syntax policy and test its behavior before you commit it, thus reducing the risk of configuration errors.

Updated: 2013-09-02

Following are a few characteristics of both classic and default syntax policies:

Name.

Each policy has a unique name.

Rule.

The rule is a logical expression that enables the NetScaler feature to evaluate a piece of traffic or another object. For example, a rule can enable the NetScaler to determine whether an HTTP request originated from a particular IP address, or whether a Cache-Control header in an HTTP request has the value “No-Cache.”

Default syntax policies can use all of the expressions that are available in a classic policy, with the exception of classic expressions for the SSL VPN client. In addition, default syntax policies enable you to configure more complex expressions.

Bindings.

To ensure that the NetScaler can invoke a policy when it is needed, you associate the policy, or bind it, to one or more bind points.
You can bind a policy globally or to a virtual server. For more information, see "About Policy Bindings."

An associated action.

An action is a separate entity from a policy. Policy evaluation ultimately results in the NetScaler performing an action. For example, a policy in the integrated cache can identify HTTP requests for .gif or .jpeg files. An action that you associate with this policy determines that the responses to these types of requests are served from the cache.

For some features, you configure actions as part of a more complex set of instructions known as a profile. For more information, see "Order of Evaluation Based on Traffic Flow."

Updated: 2013-09-30

The NetScaler supports a variety of features that rely on policies for operation. The following table summarizes how the NetScaler features use policies.

Table 1. NetScaler Feature, Policy Type, and Policy Usage

Feature Name	Policy Type	How You Use Policies in the Feature
System	Classic	For the Authentication function, policies contain authentication schemes for different authentication methods. For example, you can configure LDAP and certificate-based authentication schemes. You also configure policies in the Auditing function.
DNS	Default	To determine how to perform DNS resolution for requests.
SSL	Classic and Default	To determine when to apply an encryption function and add certificate information to clear text. To provide end-to-end security, after a message is decrypted, the SSL feature re-encrypts clear text and uses SSL to communicate with Web servers.

Feature Name	Policy Type	How You Use Policies in the Feature
Compression	Classic and Default	To determine what type of traffic is compressed.
Integrated Caching	Default	To determine whether HTTP responses are cacheable.
Responder	Default	To configure the behavior of the Responder function.
Protection Features	Classic	To configure the behavior of the Filter, SureConnect, and Priority Queuing functions.
Content Switching	Classic and Default	To determine what server or group of servers is responsible for serving responses, based on characteristics of an incoming request. Request characteristics include device type, language, cookies, HTTP method, content type, and associated cache server.
AAA - Traffic Management	Classic Exceptions: • Traffic policies support only default syntax policies • Authorization policies support both classic and default syntax policies.	To check for client-side security before users log in and establish a session. Traffic policies, which determine whether single sign-on (SSO) is required, use only the default syntax. Authorization policies authorize users and groups that access intranet resources through the appliance.
Cache Redirection	Classic	To determine whether responses are served from a cache or from an origin server.
Rewrite	Default	To identify HTTP data that you want to modify before serving. The policies provide rules for modifying the data. For example, you can modify HTTP data to redirect a request to a new home page, or a new server, or a selected server based on the address of the incoming request, or you can modify the data to mask server information in a response for security purposes. The URL Transformer function identifies URLs in HTTP transactions and text files for the purpose of evaluating whether a URL should be transformed.

Feature Name	Policy Type	How You Use Policies in the Feature
Application Firewall	Classic and Default	To identify characteristics of traffic and data that should or should not be admitted through the firewall.
NetScaler Gateway, Clientless Access function	Default	To define rewrite rules for general Web access using the NetScaler Gateway.
NetScaler Gateway	Classic	To determine how the NetScaler Gateway performs authentication, authorization, auditing, and other functions.

Updated: 2013-09-30

Policies do not themselves take action on data. Policies provide read-only logic for evaluating traffic. To enable a feature to perform an operation based on a policy evaluation, you configure actions or profiles and associate them with policies.

Note: Actions and profiles are specific to particular features. For information about assigning actions and profiles to features, see the documentation for the individual features.

About Actions

Actions are steps that the NetScaler takes, depending on the evaluation of the expression in the policy. For example, if an expression in a policy matches a particular source IP address in a request, the action that is associated with this policy determines whether the connection is permitted.

The types of actions that the NetScaler can take are feature specific. For example, in Rewrite, actions can replace text in a request, change the destination URL for a request, and so on. In Integrated Caching, actions determine whether HTTP responses are served from the cache or an origin server.

In some NetScaler features actions are predefined, and in others they are configurable. In some cases, (for example, Rewrite), you configure the actions using the same types of expressions that you use to configure the associated policy rule.

About Profiles

Some NetScaler features enable you to associate profiles, or both actions and profiles, with a policy. A profile is a collection of settings that enable the feature to perform a complex function. For example, in the application firewall, a profile for XML data can perform multiple screening operations, such as examining the data for illegal XML syntax or evidence of SQL injection.

Use of Actions and Profiles in Particular Features

The following table summarizes the use of actions and profiles in different NetScaler features. The table is not exhaustive. For more information about specific uses of actions and profiles for a feature, see the documentation for the feature.

Table 2. Use of Actions and Profiles in Different NetScaler Features

Feature	Use of an Action	Use of a Profile
Application firewall	Synonymous with a profile	All application firewall features use profiles to define complex behaviors, including pattern-based learning. You add these profiles to policies.
NetScaler Gateway	The following features of the NetScaler Gateway use actions: • **Pre-Authentication.** Uses Allow and Deny actions. You add these actions to a profile. • **Authorization.** Uses Allow and Deny actions. You add these actions to a policy. • **TCP Compression.** Uses various actions. You add these actions to a policy.	The following features use a profile: • Pre-Authentication • Session • Traffic • Clientless Access After configuring the profiles, you add them to policies.
Rewrite	You configure URL rewrite actions and add them to a policy.	Not used.
Integrated Caching	You configure caching and invalidation actions within a policy	Not used.
AAA - Traffic Management	You select an authentication type, set an authorization action of ALLOW or DENY, or set auditing to SYSLOG or NSLOG.	You can configure session profiles with a default timeout and authorization action.
Protection Features	You configure actions within policies for the following functions: • Filter • Compression • Responder • SureConnect	Not used.
SSL	You configure actions within SSL policies	Not used.
System	The action is implied. For the Authentication function, it is either Allow or Deny. For Auditing, it is Auditing On or Auditing Off.	Not used.

Feature	Use of an Action	Use of a Profile
DNS	The action is implied. It is either Drop Packets or the location of a DNS server.	Not used.
SSL Offload	The action is implied. It is based on a policy that you associate with an SSL virtual server or a service.	Not used.
Compression	Determine the type of compression to apply to the data	Not used.
Content Switching	The action is implied. If a request matches the policy, the request is directed to the virtual server associated with the policy.	Not used.
Cache Redirection	The action is implied. If a request matches the policy, the request is directed to the origin server.	Not used.

Updated: 2013-09-30

A policy is associated with, or bound to, an entity that enables the policy to be invoked. For example, you can bind a policy to request-time evaluation that applies to all virtual servers. A collection of policies that are bound to a particular bind point constitutes a policy bank.

Following is an overview of different types of bind points for a policy:

Request time global.

A policy can be available to all components in a feature at request time.

Response time global.

A policy can be available to all components in a feature at response time.

Request time, virtual server-specific.

A policy can be bound to request-time processing for a particular virtual server. For example, you can bind a request-time policy to a cache redirection virtual server to ensure that particular requests are forwarded to a load balancing virtual server for the cache, and other requests are sent to a load balancing virtual server for the origin.

Response time, virtual server-specific.

A policy can also be bound to response-time processing for a particular virtual server.

User-defined policy label.

For default syntax policies, you can configure custom groupings of policies (policy banks) by defining a policy label and collecting a set of related policies under the policy label.

Other bind points.

The availability of additional bind points depends on type of policy (classic or default syntax), and specifics of the relevant NetScaler feature. For example, classic policies that you configure for the NetScaler Gateway have user and group bind points.

For additional information about default syntax policy bindings, see "Binding Policies That Use the Default Syntax" and Configuring a Policy Bank for a Virtual Server. For additional information about classic policy bindings, see "Configuring a Classic Policy."

For classic policies, policy groups and policies within a group are evaluated in a particular order, depending on the following:

- The bind point for the policy, for example, whether the policy is bound to request-time processing for a virtual server or global response-time processing. For example, at request time, the NetScaler evaluates all request-time classic policies before evaluating any virtual server-specific policies.
- The priority level for the policy. For each point in the evaluation process, a priority level that is assigned to a policy determines the order of evaluation relative to other policies that share the same bind point. For example, when the NetScaler evaluates a bank of request-time, virtual server-specific policies, it starts with the policy that is assigned to the lowest priority value. In classic policies, priority levels must be unique across all bind points.

For default syntax policies, as with classic policies, the NetScaler selects a grouping, or bank, of policies at a particular point in overall processing. Following is the order of evaluation of the basic groupings, or banks, of default syntax policies:

1. Request-time global override
2. Request-time, virtual server-specific (one bind point per virtual server)
3. Request-time global default
4. Response-time global override
5. Response-time virtual server-specific
6. Response-time global default

However, within any of the preceding banks of policies, the order of evaluation is more flexible than in classic policies. Within a policy bank, you can point to the next policy to be evaluated regardless of the priority level, and you can invoke policy banks that belong to other bind points and user-defined policy banks.

As traffic flows through the NetScaler and is processed by various features, each feature performs policy evaluation. Whenever a policy matches the traffic, the NetScaler stores the action and continues processing until the data is about to leave the NetScaler. At that point, the NetScaler typically applies all matching actions. Integrated Caching, which only applies a final Cache or NoCache action, is an exception.

Some policies affect the outcome of other policies. Following are examples:

- If a response is served from the integrated cache, some other NetScaler features do not process the response or the request that initiated it.
- If the Content Filtering feature prevents a response from being served, no subsequent features evaluate the response.

If the application firewall rejects an incoming request, no other features can process it.

Classic and Default Syntax Expressions

May 25, 2015

One of the most fundamental components of a policy is its rule. A policy rule is a logical expression that enables the policy to analyze traffic. Most of the policy's functionality is derived from its expression.

An expression matches characteristics of traffic or other data with one or more parameters and values. For example, an expression can enable the NetScaler to accomplish the following:

- Determine whether a request contains a certificate.
- Determine the IP address of a client that sent a TCP request.
- Identify the data that an HTTP request contains (for example, a popular spreadsheet or word processing application).
- Calculate the length of an HTTP request.

This document includes the following details:

Classic expressions enable you to evaluate basic characteristics of data. They have a structured syntax that performs string matching and other operations.

Following are a few simple examples of classic expressions:

- An HTTP response contains a particular type of Cache Control header.

res.http.header Cache-Control contains public

- An HTTP response contains image data.

res.http.header Content-Type contains image/

- An SSL request contains a certificate.

req.ssl.client.cert exists

Updated: 2013-09-02

Any feature that uses default syntax policies also uses default syntax expressions. For information about which features use default syntax policies, see the table "NetScaler Feature, Policy Type, and Policy Usage."

Default syntax expressions have a few other uses. In addition to configuring default syntax expressions in policy rules, you configure default syntax expressions in the following situations:

Integrated Caching:

You use default syntax expressions to configure a selector for a content group in the integrated cache.

Load Balancing:

You use default syntax expressions to configure token extraction for a load balancing virtual server that uses the TOKEN

method for load balancing.

Rewrite:

You use default syntax expressions to configure rewrite actions.

Rate-based policies:

You use default syntax expressions to configure limit selectors when configuring a policy to control the rate of traffic to various servers.

Following are a few simple examples of default syntax expressions:

- An HTTP request URL contains no more than 500 characters.

```
http.req.url.length <= 500
```

- An HTTP request contains a cookie that has fewer than 500 characters.

```
http.req.cookie.length < 500
```

- An HTTP request URL contains a particular text string.

```
http.req.url.contains(".html")
```

Converting Classic Expressions to the Newer Default Expression Syntax

Jun 04, 2015

You can convert a classic expression to the default expression syntax by using the nspepi conversion tool. You can also use the tool to convert all the classic expressions in the NetScaler configuration to the default syntax (with the exception of NetScaler entities that currently support only classic expressions).

The conversion tool does not convert policies configured for the following features, because the features currently support only classic policies:

- Authentication, Pre-authentication
- SSL
- Cache redirection
- VPN (session, traffic, and tunnel traffic)
- Content filtering (The responder feature not only provides you with functionality that is equivalent to that provided by the content filtering feature but also surpasses the content filtering feature in the use cases that it supports. Additionally, responder supports the more powerful default syntax for policy expressions.)

The following NetScaler features support both classic and default syntax expressions and, therefore, support the conversion of classic expressions to default syntax expressions:

- Application firewall policies
- Authorization policies
- Named expressions
- Compression policies
- Content switching policies
- User-defined, rule-based tokens/persistency (the –rule parameter value that is specified for a load balancing virtual server)

This document includes the following details:

Updated: 2013-09-02

When parsing a NetScaler configuration file, the conversion tool performs the following actions:

1. In commands that create classic named expressions, the conversion tool replaces the names of the classic expressions with default syntax expressions.
2. In commands that support only the classic syntax, if classic named expressions are used, the conversion tool replaces the names of the classic expressions with the actual classic expressions they represent. This action ensures that the names of expressions in classic-only features do not reference the default syntax expressions created from Step 1.
3. In commands associated with entities that support both the classic syntax and the default syntax, the conversion tool replaces all classic expressions in commands with default syntax expressions.

Example

Consider the following sample configuration commands:

```
add policy expression ne_c1 "METHOD == GET"
add policy expression ne_c2 "ne_c1 || URL == /*.htm "
add filter policy pol1 -rule "ne_c2" -reqAction YES
add cmp policy pol2 -rule "REQ.HTTP.HEADER Accept CONTAINS \'text/html\'" -resAction COMPRESS
add cmp policy pol3 -rule "ne_c1 || ne_c2" -resAction GZIP
```

In the commands that create the classic named expressions ne_c1 and ne_c2, the tool replaces the names of the expressions with actual default syntax expressions. This action, which corresponds to Step 1 described earlier, results in the following commands:

```
add policy expression ne_c1 "HTTP.REQ.METHOD.EQ(\"GET\")"

add policy expression ne_c2 "HTTP.REQ.URL.SUFFIX.EQ(\"htm\")"
```

The filter policy command supports only the classic syntax. Therefore, the conversion tool replaces the classic named expression ne_c1 with the actual classic expression it represents. Note that the tool replaces ne_c1 in the expression for ne_c2, and then replaces ne_c2 in the filter policy with the classic expression. This action, which corresponds to Step 2 described earlier, results in the following command:

```
add filter policy pol1 -rule "METHOD == GET || URL == /*.htm" -reqAction YES
```

The compression feature supports both classic and default syntax expressions. Therefore, in the command that creates the compression policy pol2, the conversion tool replaces the expression with a default syntax expression. This action, which corresponds to Step 3 described earlier, results in the following command:

```
add cmp policy pol2 -rule "HTTP.REQ.HEADER(\"Accept\").AFTER_STR(\"text/html\").LENGTH.GT(0)" -
resAction COMPRESS
```

The command that creates the compression policy pol3 is unaffected by the conversion process because, after the conversion process is complete, ne_c1 and ne_c2 reference the default syntax expressions that result from Step 1.

Client security messages are not supported in the newer default policy format and, therefore, are lost. The SYS.EVAL_CLASSIC_EXPR function is replaced with a default policy expression. The following entities support the SYS.EVAL_CLASSIC_EXPR function:

- DNS policies
- Rate limit selectors
- Cache selectors
- Cache policies
- Content switching policies
- Rewrite policies
- URL transformation policies
- Responder policies
- Application firewall policies
- Authorization policies
- Compression policies
- CVPN access policies

After performing the conversion, the tool saves the changes in a new configuration file. The new configuration file is created in the directory in which the input file exists. The name of the new configuration file is the same as the name of the input configuration file except for the string new_ used as a prefix. Conversion warnings are reported in a warning line at the end of the screen output. Additionally, a warning file is created in the directory in which the input configuration file resides. For more

information about the warning file and the types of warnings that are reported, see "Conversion Warnings."

Updated: 2013-09-02

You can use the nspepi tool to convert a single classic expression to the default syntax. The nspepi tool must be run from the shell prompt on the NetScaler appliance.

To convert a classic expression to the default syntax by using the command line interface

At the shell prompt, type:

nspepi -e "<classic expression>"
Example

```
root@NS# nspepi -e "REQ.HTTP.URL == /*.htm"
 "HTTP.REQ.URL.REGEX_MATCH(re#/(.*)\.htm#)"
```

Updated: 2013-09-02

You can use the nspepi tool to convert all the classic expressions in a NetScaler configuration file to the default syntax (except for those commands that do not support the default syntax). The nspepi tool must be run from the shell prompt on the NetScaler appliance.

To convert all the classic expressions in a NetScaler configuration file to the default syntax by using the command line interface

At the shell prompt, type:

nspepi -f "<ns config file>" -v
Example

```
root@NS# nspepi -f ns.conf
OUTPUT: New configuration file created: new_ns.conf
OUTPUT: New warning file created: warn_ns.conf
WARNINGS: Total number of warnings due to bind commands: 18
WARNINGS: Line numbers which has bind command issues: 305, 306, 706, 707, 708, 709, 710, 711, 712, 713,
714, 715, 767, 768, 774, 775, 776, 777
root@NS#
```

When classic expressions that are included in CLI commands are upgraded to the default syntax, the number of characters in the expression might exceed the 1499-character limit. The commands that include expressions longer than 1499 characters fail when the configuration is being applied. You must manually update these commands.

In addition, multiple classic policies can be bound to a given bind point with priority 0 or with equal priority, but the default syntax policy infrastructure does not support a priority value of 0 or policies with the same priority at a given bind point. These commands fail when the configuration is being applied. The commands must be updated manually with the correct priority values.

The line numbers of lines that threw a warning during conversion are listed at the end of the output in a warning line. In addition, a warning file is created in the same directory as the one in which the old and new configuration files reside. The name of the warning file is the same as the name of the input configuration file except that the string **warn_** is added as a prefix.

Before You Proceed

May 25, 2015

Before configuring expressions and policies, be sure you understand the relevant NetScaler feature and the structure of your data, as follows:

- Read the documentation on the relevant feature.
- Look at the data stream for the type of data that you want to configure.

You may want to run a trace on the type of traffic or content that you want to configure. This will give you an idea of the parameters and values, and operations on these parameters and values, that you need to specify in an expression.

Note: The NetScaler supports either classic or default syntax policies within a feature. You cannot have both types in the same feature. Over the past few releases, some NetScaler features have migrated from using classic policies and expressions to default syntax policies and expressions. If a feature of interest to you has changed to the default syntax format, you may have to manually migrate the older information. Following are guidelines for deciding if you need to migrate your policies:

- If you configured classic policies in a version of the Integrated Caching feature prior to release 9.0 and then upgrade to version 9.0 or later, there is no impact. All legacy policies are migrated to the default syntax policy format.
- For other features, you need to manually migrate classic policies and expressions to the default syntax if the feature has migrated to the default syntax.

Configuring Default Syntax Policies

Sep 30, 2013

You can create default syntax policies for various NetScaler features, including DNS, Rewrite, Responder, and Integrated Caching, and the clientless access function in the NetScaler Gateway. Policies control the behavior of these features.

When you create a policy, you assign it a name, a rule (an expression), feature-specific attributes, and an action that is taken when data matches the policy. After creating the policy, you determine when it is invoked by binding it globally or to either request-time or response-time processing for a virtual server.

Policies that share the same bind point are known as a *policy bank*. For example, all policies that are bound to a virtual server constitute the policy bank for the virtual server. When binding the policy, you assign it a priority level to specify when it is invoked relative to other policies in the bank. In addition to assigning a priority level, you can configure an arbitrary evaluation order for policies in a bank by specifying Goto expressions.

In addition to policy banks that are associated with a built-in bind point or a virtual server, you can configure *policy labels*. A policy label is a policy bank that is identified by an arbitrary name. You invoke a policy label, and the policies in it, from a global or virtual-server-specific policy bank. A policy label or a virtual-server policy bank can be invoked from multiple policy banks.

For some features, you can use the policy manager to configure and bind policies.

Rules for Names in Identifiers Used in Policies

Mar 20, 2012

The names of identifiers in the named expression, HTTP callout, pattern set, and rate limiting features must begin with an ASCII alphabet or an underscore (_). The remaining characters can be ASCII alphanumeric characters or underscores (_).

The names of these identifiers must not begin with the following reserved words:

- The words ALT, TRUE, or FALSE or the Q or S one-character identifier.
- The special-syntax indicator RE (for regular expressions) or XP (for XPath expressions).
- Expression prefixes, which currently are the following:
  - CLIENT
  - EXTEND
  - HTTP
  - SERVER
  - SYS
  - TARGET
  - TEXT
  - URL
  - MYSQL
  - MSSQL

Additionally, the names of these identifiers cannot be the same as the names of enumeration constants used in the policy infrastructure. For example, the name of an identifier cannot be IGNORECASE, YEAR, or LATIN2_CZECH_CS (a MySQL character set).

Note: The NetScaler appliance performs a case-insensitive comparison of identifiers with these words and enumeration constants. For example, names of the identifiers cannot begin with TRUE, True, or true.

Creating or Modifying a Policy

Nov 14, 2013

All policies have some common elements. Creating a policy consists, at minimum, of naming the policy and configuring a rule. The policy configuration tools for the various features have areas of overlap, but also differences. For the details of configuring a policy for a particular feature, including associating an action with the policy, see the documentation for the feature.

To create a policy, begin by determining the purpose of the policy. For example, you may want to define a policy that identifies HTTP requests for image files, or client requests that contain an SSL certificate. In addition to knowing the type of information that you want the policy to work with, you need to know the format of the data that the policy is analyzing.

Next, determine whether the policy is globally applicable, or if it pertains to a particular virtual server. Also consider the effect that the order in which your policies are evaluated (which will be determined by how you bind the policies) will have on the policy that you are about to configure.

At the command prompt, type the following commands to create a policy and verify the configuration:

- add responder|dns|cs|rewrite|cache policy <policyName> -rule <expression> [<feature-specific information>]
- show rewrite policy <name>

Example 1:

```
add rewrite policy "pol_remove-ae" true "act_remove-ae"
 Done
> show rewrite policy pol_remove-ae
      Name: pol_remove-ae
      Rule: true
      RewriteAction: act_remove-ae
      UndefAction: Use Global
      Hits: 0
      Undef Hits: 0
      Bound to: GLOBAL RES_OVERRIDE
      Priority: 90
      GotoPriorityExpression: END
 Done
>
```

Example 2:

```
add cache policy BranchReportsCachePolicy -rule q{http.req.url.query.value("actionoverride").contains("branchReport s")} -action cache
Done
 show cache policy BranchReportsCachePolicy
      Name: BranchReportsCachePolicy
      Rule: http.req.url.query.value("actionoverride").contains("branchReports")
      CacheAction: CACHE
      Stored in group: DEFAULT
      UndefAction: Use Global
      Hits: 0
      Undef Hits: 0
 Done
```

Note: At the command line, quote marks within a policy rule (the expression) must be escaped or delimited with the q delimiter. For more information, see "Configuring Default Syntax Expressions in a Policy."

1. In the navigation pane, expand the name of the feature for which you want to configure a policy, and then click Policies. For example, you can select Content Switching, Integrated Caching, DNS, Rewrite, or Responder.
2. In the details pane, click Add, or select an existing policy and click Open. A policy configuration dialog box appears.
3. Specify values for the following parameters. (An asterisk indicates a required parameter. For a term in parentheses, see the corresponding parameter in "Parameters for creating or modifying a policy.")
4. Click Create, and then click Close.
5. Click Save. A policy is added.
   Note: After you create a policy, you can view the policy's details by clicking the policy entry in the configuration pane. Details that are highlighted and underlined are links to the corresponding entity (for example, a named expression).

Policy Configuration Examples

Sep 02, 2013

These examples show how policies and their associated actions are entered at the command line interface. In the configuration utility, the expressions would appear in the Expression window of the feature-configuration dialog box for the integrated caching or rewrite feature.

Following is an example of creating a caching policy. Note that actions for caching policies are built in, so you do not need to configure them separately from the policy.

```
add cache policy BranchReportsCachePolicy -rule q{http.req.url.query.value("actionoverride").contains("branchReports")} -action cache
```

Following is an example of a Rewrite policy and action:

```
add rewrite action myAction1 INSERT_HTTP_HEADER "myHeader" "valueForMyHeader"
add rewrite policy myPolicy1 "http.req.url.contains(\"myURLstring\")" myAction1
```

Note: At the command line, quote marks within a policy rule (the expression) must be escaped or delimited with the q delimiter. For more information, see "Configuring Default Syntax Expressions in a Policy."

Binding Policies That Use the Default Syntax

Sep 30, 2013

After defining a policy, you indicate when the policy is to be invoked by binding the policy to a bind point and specifying a priority level. You can bind a policy to only one bind point. A bind point can be global, that is, it can apply to all virtual servers that you have configured. Or, a bind point can be specific to a particular virtual server, which can be either a load balancing or a content switching virtual server. Not all bind points are available for all features.

The order in which policies are evaluated determines the order in which they are applied, and features typically evaluate the various policy banks in a particular order. Sometimes, however, other features can affect the order of evaluation. Within a policy bank, the order of evaluation depends on the values of parameters configured in the policies. Most features apply all of the actions associated with policies whose evaluation results in a match with the data that is being processed. The integrated caching feature is an exception.

Feature-Specific Differences in Policy Bindings

You can bind policies to built-in, global bind points (or banks), to virtual servers, or to policy labels.

However, the NetScaler features differ in terms of the types of bindings that are available. The following table summarizes how you use policy bindings in various NetScaler features that use policies.

Table 1. Feature-Specific Bindings for Policies

Feature Name	Virtual Servers Configured in the Feature	Policies Configured in the Feature	Bind Points Configured for the Policies	Use of Policies in the Feature
DNS	none	DNS policies	Global	To determine how to perform DNS resolution for requests.
Content Switching Note: This feature can support either or classic policies or policies that use the default syntax, but not both.	Content Switching (CS)	Content Switching policies	• Content switching or cache redirection virtual server • Policy label	To determine what server or group of servers is responsible for serving responses, based on characteristics of an incoming request. Request characteristics include device type, language, cookies, HTTP method, content type, and associated cache server.
Integrated Caching	none	Caching policies	• Global override • Global default	To determine whether HTTP responses can be stored in, and served from, the NetScaler appliance's integrated cache.

Feature Name	Virtual Servers Configured in the Feature	Policies Configured in the Feature	Bind Points Configured for the Policies	Use of Policies in the Feature
			• Policy label • Load balancing, content switching, or SSL offload virtual server	
Responder	none	Responder policies	• Global override • Global default • Policy label • Load balancing, content switching, or SSL offload virtual server	To configure the behavior of the Responder function.
Rewrite	none	Rewrite policies	• Global override • Global default • Policy label • Load balancing, content switching, or SSL offload virtual server	To identify HTTP data that you want to modify before serving. The policies provide rules for modifying the data. For example, you can modify HTTP data to redirect a request to a selected server based on the address of the incoming request, or to mask server information in a response for security purposes.
URL Transform function in the Rewrite feature	none	Transformation policies	• Global override • Global default	To identify URLs in HTTP transactions and text files for the purpose of evaluating whether a URL should be altered.

Feature Name	Virtual Servers Configured in the Feature	Policies Configured in the Feature	Bind Points Configured for the Policies	Use of Policies in the Feature
NetScaler Gateway (clientless VPN functions only)	VPN server	Clientless Access policies	• Policy label • VPN Global • VPN server	To determine how the NetScaler Gateway performs authentication, authorization, auditing, and other functions, and to define rewrite rules for general Web access using the NetScaler Gateway.

Bind Points and Order of Evaluation

For a policy to take effect, you must ensure that the policy is invoked at some point during processing. To do so, you associate the policy with a bind point. The collection of policies that is bound to a bind point is known as a policy bank.

Following are the bind points that the NetScaler evaluates, listed in the typical order of evaluation within a policy bank

1. **Request-time override.** When a request flows through a feature, the NetScaler first evaluates request-time override policies for the feature.
2. **Request-time Load Balancing virtual server.** If policy evaluation cannot be completed after all the request-time override policies have been evaluated, the NetScaler processes request-time policies for load balancing virtual servers.
3. **Request-time Content Switching virtual server.** If policy evaluation cannot be completed after all the request-time policies for load balancing virtual servers have been evaluated, the NetScaler processes request-time policies for content switching virtual servers.
4. **Request-time default.** If policy evaluation cannot be completed after all request-time, virtual server-specific policies have been evaluated, the NetScaler processes request-time default policies.
5. **Response-time override.** At response time, the NetScaler starts with policies that are bound to the response-time override bind point.
6. **Response-time Load Balancing virtual server.** If policy evaluation cannot be completed after all response-time override policies have been evaluated, the NetScaler process the response-time policies for load balancing virtual servers.
7. **Response-time Content Switching virtual server.** If policy evaluation cannot be completed after all policies have been evaluated for load balancing virtual servers, the NetScaler process the response-time policies for content switching virtual servers.
8. **Response-time default.** If policy evaluation cannot be completed after all response-time, virtual-server-specific policies have been evaluated, the NetScaler processes response-time default policies.

Policy Evaluation across Features

In addition to attending to evaluation of policies within a feature, if you have bound policies to a content switching virtual server, note that these policies are evaluated before other policies. Binding a policy to a content switching vserver produces a different result in NetScaler versions 9.0.x and later than in 8.x versions. In NetScaler 9.0 and later versions, evaluation occurs as follows:

- Content switching policies are evaluated before other policies. If a content switching policy evaluates to TRUE, the target load balancing vserver is selected.
- If all content switching policies evaluate to FALSE, the default load balancing vserver under the content switching VIP is selected.

After a target load balancing vserver is selected by the content switching process, policies are evaluated in the following

order:

1. Policies that are bound to the global override bind point.
2. Policies that are bound to the default load balancing vserver.
3. Policies that are bound to the target content switching vserver.
4. Policies that are bound to the global default bind point.

To be sure that the policies are evaluated in the intended order, follow these guidelines:

- Make sure that the default load balancing vserver is not directly reachable from the outside; for example, the vserver IP address can be 0.0.0.0.
- To prevent exposing internal data on the load balancing default vserver, configure a policy to respond with a "503 Service Unavailable" status and bind it to the default load balancing vserver.

Entries in a Policy Bank

Each entry in a policy bank has, at minimum, a policy and a priority level. You can also configure entries that change the priority-based evaluation order, and you can configure entries that invoke external policy banks.

The following table summarizes each entry in a policy bank.

Table 2. Format of Each Entry in a Policy Bank

Policy Name	Priority	Goto Expression	Invocation Type	Policy Bank to Be Invoked
The policy name, or a "dummy" policy named NOPOLICY. The NOPOLICY entry controls evaluation flow without processing a rule.	An integer.	Optional. Identifies the next policy in the bank to evaluate, or ends any further evaluation	Optional. Indicates that an external policy bank will be invoked. This field restricts the choices to a global policy label or a virtual server.	Optional. Used with Invocation Type. This is the label for a policy bank or a virtual server name. The NetScaler returns to the current bank after processing the external bank.

If the policy evaluates to TRUE, the NetScaler stores the action that is associated with the policy. If the policy evaluates to FALSE, the NetScaler evaluates the next policy. If the policy is neither TRUE nor FALSE, the NetScaler uses the associated Undef (undefined) action.

Evaluation Order within a Policy Bank

Within a policy bank, the evaluation order depends on the following items:

A priority.

The most minimal amount of information about evaluation order is a numeric priority level. The lower the number, the higher the priority.

A Goto expression.

If supplied, the Goto expression indicates the next policy to be evaluated, typically within the same policy bank.. Goto expressions can only proceed forward in a bank. To prevent looping, a policy bank configuration is not valid if a Goto statement points backwards in the bank.

Invocation of other policy banks.

Any entry can invoke an external policy bank. The NetScaler provides a built-in entity named NOPOLICY that does not have a rule. You can add a NOPOLICY entry in a policy bank when you want to invoke another policy bank, but do not want to process any other rules prior to the invocation. You can have multiple NOPOLICY entries in multiple policy banks.

Values for a Goto expression are as follows:

NEXT.

This keyword selects the policy with the next higher priority level in the current policy bank.

An integer.

If you supply an integer, it must match the priority level of another policy in the current policy bank.

END.

This keyword stops evaluation after processing the current policy, and no additional policies in this bank are processed.

Blank.

If the Goto expression is empty, it is the same as specifying END.

A numeric expression.

This is a default syntax expression that resolves to a priority number for another policy in the current bank.

USE_INVOCATION_RESULT.

This phrase can be used only if you are invoking an external policy bank. Entering this phrase causes the NetScaler to perform one of the following actions:

- If the final Goto in the invoked policy bank has a value of END or is empty, the invocation result is END, and evaluation stops.
- If the final Goto expression in the invoked policy bank is anything other than END, the NetScaler performs a NEXT.

The following table illustrates a policy bank that uses Goto statements and policy bank invocations.

Table 3. Example of a Policy Bank That Uses Gotos and External Bank Invocations

Policy Name	Priority	Goto	Invocation	Policy Bank to Be Invoked
ClientCertificatePolicy (rule: does the request contain a client certificate?)	100	300	None	None
SubnetPolicy (rule: is the client from a private subnet?)	200	NEXT	None	None
NOPOLICY	300	USE INVOCATION RESULT	Request vserver	My_Request_VServer

Policy Name	Priority	Goto INVOCATION RESULT	Policy Invocation Label	Policy Bank to Be Invoked
NOPOLICY	350	USE	Policy	My_Policy_Label
WorkingHoursPolicy (rule: is it working hours?)	400	END	None	None

How Policy Evaluation Ends

Evaluation of a policy bank ends when one of the following takes place:

- A policy evaluates to TRUE and its Goto statement value is END.
  No further policies or policy banks in this feature are evaluated.
- An external policy bank is invoked, its evaluation returns an END, and the Goto statement uses a value of USE_INVOCATION_RESULT or END.
  Evaluation continues with the next policy bank for this feature. For example, if the current bank is the request-time override bank, the NetScaler next evaluates request-time policy banks for the virtual servers.
- The NetScaler has walked through all the policy banks in this feature, but has not encountered an END.
  If this is the last entry to be evaluated in this policy bank, the NetScaler proceeds to the next feature.

How Features Use Actions after Policy Evaluation

After evaluating all relevant policies for a particular data point (for example, an HTTP request), the NetScaler stores all the actions that are associated with any policy that matched the data.

For most features, all the actions from matching policies are applied to a traffic packet as it leaves the NetScaler. The Integrated Caching feature only applies one action: CACHE or NOCACHE. This action is associated with the policy with the lowest priority value in the “highest priority” policy bank (for example, request-time override policies are applied before virtual server-specific policies).

Binding a Policy Globally

Nov 14, 2013

The following binding procedures are typical. However, refer to the documentation for the feature of interest to you for complete instructions.

To bind an Integrated Caching policy globally by using the command line interface

At the command prompt, type the following commands to bind an Integrated Caching policy and verify the configuration:

- bind cache global <policy> -priority <positiveInteger> [-type REQ_OVERRIDE | REQ_DEFAULT | RES_OVERRIDE | RES_DEFAULT]
- show cache global

  Example

```
bind cache global _nonPostReq -priority 100 -type req_default
 Done
> show cache global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 2

2)     Global bindpoint: RES_DEFAULT
       Number of bound policies: 1
 Done
```

  The type argument is optional to maintain backward compatibility. If you omit the type, the policy is bound to REQ_DEFAULT or RES_DEFAULT, depending on whether the policy rule is a response-time or a request-time expression.

To bind a Rewrite policy globally by using the command line interface

At the command prompt, type the following commands to bind a Rewrite policy and verify the configuration:

- bind rewrite global <policyName> <priority> [-type REQ_OVERRIDE | REQ_DEFAULT | RES_OVERRIDE | RES_DEFAULT]
- show rewrite global

  Example

```
bind rewrite global pol_remove-pdf 100
 Done
> show rewrite global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 1

2)     Global bindpoint: REQ_OVERRIDE
       Number of bound policies: 1

 Done
```

  The type argument is optional for globally bound policies, to maintain backward compatibility. If you omit the type, the policy is bound to REQ_DEFAULT or RES_DEFAULT, depending on whether the policy rule is a response-time or a request-time expression.

To bind a compression policy globally by using the command line interface

At the command prompt, type the following commands to bind a compression policy and verify the configuration:

- bind cmp global <policyName> -priority <positiveInteger> [-type REQ_OVERRIDE | REQ_DEFAULT | RES_OVERRIDE | RES_DEFAULT]
- show cmp global

Example

```
> bind cmp global cmp_pol_1 -priority 100
 Done
> show cmp policy cmp_pol_1
        Name: cmp_pol_1
        Rule: HTTP.REQ.URL.SUFFIX.EQ("BMP")
        Response Action: COMPRESS
        Hits: 0

        Policy is bound to following entities
        1) GLOBAL REQ_DEFAULT
        Priority: 100
        GotoPriorityExpression: END
 Done
>
```

To bind a Responder policy globally by using the command line interface

At the command prompt, type the following commands to bind a Responder policy and verify the configuration:

- bind responder global <policyName> <priority> [-type OVERRIDE | DEFAULT]
- show responder global

Example

```
 bind responder global pol404Error1 200
 Done
> show responder global
1)     Global bindpoint: REQ_DEFAULT
        Number of bound policies: 1


 Done
```

To bind a DNS policy globally by using the command line interface

At the command prompt, type the following commands to bind a DNS policy and verify the configuration:

- bind dns global <policyName> <priority>
- show dns global

Example

```
> bind dns global pol_ddos_drop1 150
 Done
> show dns global
Policy name : pol_ddos_drop
        Priority : 100
```

```
        Goto expression : END
    Policy name : pol_ddos_drop1
        Priority : 150
     Done
    >
```

To bind an Integrated Caching, Responder, Rewrite, or Compression policy globally by using the configuration utility

1. In the navigation pane, click the name of the feature for which you want to bind the policy.
2. In the details pane, click <Feature Name> policy manager.
3. In the Policy Manager dialog box, select the bind point to which you want to bind the policy (for example, for Integrated Caching, Rewrite, or Compression, you could select Request and Default Global). The Responder does not differentiate between request-time and response-time policies.
4. Click Insert Policy and, from the Policy Name pop-up menu, select the policy name. A priority is assigned automatically to the policy, but you can click the cell in the Priority column and drag it anywhere within the dialog box if you want the policy to be evaluated after other policies in this bank. The priority is automatically reset. Note that priority values within a policy bank must be unique.
5. Click Apply Changes.
6. Click Close. A message in the status bar indicates that the policy is bound successfully.

To bind a DNS policy globally by using the configuration utility

1. Navigate to Traffic Management > DNS > Policies.
2. In the details pane, click Global Bindings.
3. In the global bindings dialog box, click Insert Policy, and select the policy that you want to bind globally.
4. Click in the Priority field and enter the priority level.
5. Click OK. A message in the status bar indicates that the policy is bound successfully.

Binding a Policy to a Virtual Server

Nov 14, 2013

A globally bound policy applies to all load balancing and content switching virtual servers.

Note that when binding a policy to a virtual server, you must identify it as a request-time or a response-time policy.

To bind a policy to a load balancing or content switching virtual server by using the command line interface

At the command prompt, type the following commands to bind a policy to a load balancing or content switching virtual server and verify the configuration:

- bind lb|cs vserver <virtualServerName> -policyName <policyName> -priority <positiveInteger> -type REQUEST|RESPONSE
- show lb vserver <name>

Example

```
> bind lb vserver lbvip -policyName ns_cmp_msapp -priority 50
 Done
> show lb vserver lbvip
        lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
        State: DOWN
        Last state change was at Wed Jul 15 05:54:24 2009 (+226 ms)
        Time since last state change: 28 days, 01:57:26.350
        Effective State: DOWN
        Client Idle Timeout: 180 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Port Rewrite : DISABLED
        No. of Bound Services :  0 (Total)      0 (Active)
        Configured Method: LEASTCONNECTION
        Mode: IP
        Persistence: NONE
        Vserver IP and Port insertion: OFF
        Push: DISABLED  Push VServer:
        Push Multi Clients: NO
        Push Label Rule: none

    1)      Policy : ns_cmp_msapp Priority:50
    2)      Policy : cf-pol Priority:1      Inherited
     Done
```

To bind a policy to an SSL offload virtual server by using the command line interface

At the command prompt, type the following commands to bind a policy to an SSL offload virtual server and verify the configuration:

bind ssl vserver <vServerName>@ -policyName <policyName> -priority <positiveInteger>

To bind a policy to a virtual server by using the configuration utility

1. In the navigation pane, expand Traffic Management > Load Balancing, Traffic Management > Content Switching, Traffic Management > SSL Offload, Security > AAA- Application Traffic, or NetScaler Gateway, and then click Virtual Servers.
2. In the details pane, double-click the virtual server to which you want to bind the policy, and then click Open.
3. On the Policies tab, click the icon for the type of policy that you want to bind (the choices are feature-specific), and then click the name of the policy. Note that for some features, you can bind both classic policies and policies that use the default syntax to the virtual server.
4. If you are binding a policy to a Content Switching virtual server, in the Target field select a load balancing virtual server to which traffic that matches the policy is sent.
5. Click OK. A message in the status bar indicates that the policy is bound successfully.

Displaying Policy Bindings

Oct 29, 2013

You can display policy bindings to verify that they are correct.

To display policy bindings by using the command line interface

At the command prompt, type the following commands to display policy bindings and verify the configuration:
show rewrite policy <name>

Example

```
> show rewrite policy pol_remove-pdf
      Name: pol_remove-pdf
      Rule: http.req.url.contains(".pdf")
      RewriteAction: act_remove-ae
      UndefAction: Use Global
      Hits: 0
      Undef Hits: 0
      Bound to: GLOBAL REQ_DEFAULT
      Priority: 100
      GotoPriorityExpression: END
 Done
>
```

To display global policy bindings for Integrated Caching, Rewrite, or Responder by using the configuration utility

1. In the navigation pane, expand the feature that contains the policy that you want to view, and then click Policies.
2. In the details pane, click the policy. Bound policies have a check mark next to them.
3. At the bottom of the page, under Details, next to Bound to, view the entity to which the policy is bound.

To display global policy bindings for DNS or Clientless Access in the NetScaler Gateway by using the configuration utility

1. Navigate to Traffic Management > DNS > Policies.
2. In the details pane, click Global Bindings.

To display global policy bindings for Content Switching by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In detailed pane, select policy.
3. In the details pane, click Show Bindings.

Unbinding a Policy

Nov 14, 2013

If you want to re-assign a policy or delete it, you must first remove its binding.

To unbind an integrated caching, rewrite, or compression default syntax policy globally by using the command line interface

At the command prompt, type the following commands to unbind an integrated caching, rewrite, or compression default syntax policy globally and verify the configuration:

- unbind cache | rewrite | cmp global <policyName> [-type req_override | req_default | res_override | res_default] [-priority <positiveInteger>]
- show cache | rewrite | cmp global

Example

```
> unbind cache global_nonPostReq
 Done
> show cache global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 1

2)     Global bindpoint: RES_DEFAULT
       Number of bound policies: 1

 Done
```

The priority is required only for the “dummy” policy named NOPOLICY.

To unbind a responder policy globally by using the command line interface

At the command prompt, type the following commands to unbind a responder policy globally and verify the configuration:

- unbind responder global <policyName> [-type override | default] [-priority <positiveInteger>]
- show responder global

Example

```
> unbind responder global pol404Error
 Done
> show responder global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 1

 Done
```

The priority is required only for the “dummy” policy named NOPOLICY.

To unbind a DNS policy globally by using the command line interface

At the command prompt, type the following commands to unbind a DNS policy globally and verify the configuration:

- unbind responder global <policyName>
- unbind responder global

Example

```
unbind dns global dfgdfg
Done
show dns global
Policy name : dfgdfggfhg
      Priority : 100
      Goto expression : END
Done
```

To unbind a default syntax policy from a virtual server by using the command line interface

At the command prompt, type the following commands to unbind a default syntax policy from a virtual server and verify the configuration:

- unbind cs vserver <name> -policyName <policyName> [-priority <positiveInteger>] [-type REQUEST | RESPONSE]
- show lb vserver <name>

Example

```
unbind cs vserver vs-cont-switch  -policyName pol1
Done
> show cs vserver vs-cont-switch
      vs-cont-switch (10.102.29.10:80) - HTTP Type: CONTENT
      State: UP
      Last state change was at Wed Aug 19 08:56:55 2009 (+18 ms)
      Time since last state change: 0 days, 02:47:55.750
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Port Rewrite : DISABLED
      State Update: DISABLED
      Default:       Content Precedence: RULE
      Vserver IP and Port insertion: OFF
      Case Sensitivity: ON
      Push: DISABLED  Push VServer:
      Push Label Rule: none
Done
```

The priority is required only for the “dummy” policy named NOPOLICY.

To unbind an integrated caching, responder, rewrite, or compression default syntax policy globally by using the configuration utility

1. In the navigation pane, click the feature with the policy that you want to unbind (for example, Integrated Caching).
2. In the details pane, click <Feature Name> policy manager.
3. In the Policy Manager dialog box, select the bind point with the policy that you want to unbind, for example, Default Global.
4. Click the policy name that you want to unbind, and then click Unbind Policy.
5. Click Apply Changes.
6. Click Close. A message in the status bar indicates that the policy is unbound successfully.

To unbind a DNS policy globally by using the configuration utility

1. Navigate to Traffic Management > DNS > Policies.
2. In the details pane, click Global Bindings.
3. In the Global Bindings dialog box, select policy and click unbind policy.
4. Click OK. A message in the status bar indicates that the policy is unbinded successfully.

To unbind a default syntax policy from a load balancing or content switching virtual server by using the configuration utility

1. Navigate to Traffic Management, and expand Load Balancing or Content Switching, and then click Virtual Servers.
2. In the details pane, double-click the virtual server from which you want to unbind the policy.
3. On the Policies tab, in the Active column, clear the check box next to the policy that you want to unbind.
4. Click OK. A message in the status bar indicates that the policy is unbinded successfully.

Creating Policy Labels

May 25, 2015

In addition to the built-in bind points where you set up policy banks, you can also configure user-defined policy labels and associate policies with them.

Within a policy label, you bind policies and specify the order of evaluation of each policy relative to others in the bank of policies for the policy label. The NetScaler also permits you to define an arbitrary evaluation order as follows:

- You can use “goto” expressions to point to the next entry in the bank to be evaluated after the current one.
- You can use an entry in a policy bank to invoke another bank.

This document includes the following details:

Creating Policy Labels

Updated: 2013-11-14

Each feature determines the type of policy that you can bind to a policy label, the type of load balancing virtual server that you can bind the label to, and the type of content switching virtual server from which the label can be invoked. For example, a TCP policy label can only be bound to a TCP load balancing virtual server. You cannot bind HTTP policies to a policy label of this type. And you can invoke a TCP policy label only from a TCP content switching virtual server.

After configuring a new policy label, you can invoke it from one or more banks for the built-in bind points.

To create a caching policy label by using the command line interface

At the command prompt, type the following commands to create a Caching policy label and verify the configuration:

- add cache policylabel <labelName> -evaluates req|res
- show cache policylabel<labelName>

Example

```
> add cache policylabel lbl-cache-pol -evaluates req
 Done

> show cache policylabel lbl-cache-pol
      Label Name: lbl-cache-pol
      Evaluates: REQ
      Number of bound policies: 0
      Number of times invoked: 0
 Done
>
```

To create a Content Switching policy label by using the command line interface

At the command prompt, type the following commands to create a Content Switching policy label and verify the configuration:

- add cs policylabel <labelName> http|tcp|rtsp|ssl

- show cs policylabel <labelName>

  Example

```
> add cs policylabel lbl-cs-pol http
 Done
> show cs policylabel lbl-cs-pol
      Label Name: lbl-cs-pol
      Label Type: HTTP
      Number of bound policies: 0
      Number of times invoked: 0
 Done
```

To create a Rewrite policy label by using the command line interface

At the command prompt, type the following commands to create a Rewrite policy label and verify the configuration:

- add rewrite policylabel <labelName> http_req|http_res|url|text|clientless_vpn_req|clientless_vpn_res
- show rewrite policylabel <labelName>

  Example

```
> add rewrite policylabel lbl-rewrt-pol http_req
 Done

> show rewrite policylabel lbl-rewrt-pol
      Label Name: lbl-rewrt-pol
      Transform Name: http_req
      Number of bound policies: 0
      Number of times invoked: 0
 Done
```

To create a Responder policy label by using the command line interface

At the command prompt, type the following commands to create a Responder policy label and verify the configuration:

- add responder policylabel <labelName>
- show responder policylabel <labelName>

  Example

```
> add responder policylabel lbl-respndr-pol
 Done

> show responder policylabel lbl-respndr-pol
      Label Name: lbl-respndr-pol
      Number of bound policies: 0
      Number of times invoked: 0
 Done
```

Note: Invoke this policy label from a policy bank. For more information, see "Binding a Policy to a Policy Label."

To create a policy label by using the configuration utility

1. In the navigation pane, expand the feature for which you want to create a policy label, and then click Policy Labels. The choices are Integrated Caching, Rewrite, Content Switching, or Responder.
2. In the details pane, click Add.
3. In the Name box, enter a unique name for this policy label.
4. Enter feature-specific information for the policy label. For example, for Integrated Caching, in the Evaluates drop-down menu, you would select REQ if you want this policy label to contain request-time policies, or select RES if you want this policy label to contain response-time policies. For Rewrite, you would select a Transform name.
5. Click Create.
6. Configure one of the built-in policy banks to invoke this policy label. For more information, see "Binding a Policy to a Policy Label." A message in the status bar indicates that the policy label is created successfully.

Binding a Policy to a Policy Label

As with policy banks that are bound to the built-in bind points, each entry in a policy label is a policy that is bound to the policy label. As with policies that are bound globally or to a vserver, each policy that is bound to the policy label can also invoke a policy bank or a policy label that is evaluated after the current entry has been processed. The following table summarizes the entries in a policy label.

Name

The name of a policy, or, to invoke another policy bank without evaluating a policy, the “dummy” policy name NOPOLICY. You can specify NOPOLICY more than once in a policy bank, but you can specify a named policy only once.

Priority

An integer. This setting can work with the Goto expression.

Goto Expression

Determines the next policy to evaluate in this bank. You can provide one of the following values:

NEXT:

Go to the policy with the next higher priority.

END:

Stop evaluation.

USE_INVOCATION_RESULT:

Applicable if this entry invokes another policy bank. If the final Goto in the invoked bank has a value of END, evaluation stops. If the final Goto is anything other than END, the current policy bank performs a NEXT.

Positive number:

The priority number of the next policy to be evaluated.

Numeric expression:

An expression that produces the priority number of the next policy to be evaluated.

The Goto can only proceed forward in a policy bank.

If you omit the Goto expression, it is the same as specifying END.

Invocation Type

Designates a policy bank type. The value can be one of the following:

Request Vserver:

Invokes request-time policies that are associated with a virtual server.

Response Vserver:

Invokes response-time policies that are associated with a virtual server.

Policy label:

Invokes another policy bank, as identified by the policy label for the bank.

Invocation Name

The name of a virtual server or a policy label, depending on the value that you specified for the Invocation Type.

Configuring a Policy Label or Virtual Server Policy Bank

May 25, 2015

After you have created policies, and created policy banks by binding the policies, you can perform additional configuration of polices within a label or policy bank. For example, before you configure invocation of an external policy bank, you might want to wait until you have configured that policy bank.

This document includes the following details:

- Configuring a Policy Label
- Configuring a Policy Bank for a Virtual Server

Configuring a Policy Label

Updated: 2013-11-14

A policy label consists of a set of policies and invocations of other policy labels and virtual server-specific policy banks. An Invoke parameter enables you to invoke a policy label or a virtual server-specific policy bank from any other policy bank. A special-purpose NoPolicy entry enables you to invoke an external bank without processing an expression (a rule). The NoPolicy entry is a "dummy" policy that does not contain a rule.

For configuring policy labels from the NetScaler command line, note the following elaborations of the command syntax:

- gotoPriorityExpression is configured as described in "Entries in a Policy Bank."
- The type argument is required. This is unlike binding a conventional policy, where this argument is optional.
- You can invoke the bank of policies that are bound to a virtual server by using the same method as you use for invoking a policy label.

To configure a policy label by using the command line interface

At the command prompt, type the following commands to configure a policy label and verify the configuration:

- bind cache | rewrite | responder policylabel <policylabelName> -policyName <policyName> -priority <priority> [-gotoPriorityExpression <gotopriorityExpression>] [-invoke reqvserver | resvserver | policylabel <policyLabelName> | <vserverName>]
- show cache | rewrite | responder policylabel <policylabelName>

Example

```
bind cache policylabel _reqBuiltinDefaults -policyName  _nonGetReq -priority 100
Done
show cache policylabel _reqBuiltinDefaults
      Label Name: _reqBuiltinDefaults
      Evaluates: REQ
      Number of bound policies: 3
      Number of times invoked: 0
1)     Policy Name: _nonGetReq
       Priority: 100
       GotoPriorityExpression: END
2)     Policy Name: _advancedConditionalReq
      Priority: 200
      GotoPriorityExpression: END

3)     Policy Name: _personalizedReq
      Priority: 300
      GotoPriorityExpression: END
Done
```

To invoke a policy label from a Rewrite policy bank with a NOPOLICY entry by using the command line interface

At the command prompt, type the following commands to invoke a policy label from a Rewrite policy bank with a NOPOLICY entry and verify the configuration:

- bind rewrite global <policyName> <priority> <gotoPriorityExpression> -type REQ_OVERRIDE | REQ_DEFAULT | RES_OVERRIDE | RES_DEFAULT -invoke reqvserver | resvserver | policylabel <policyLabelName> | <vserverName>
- show rewrite global

Example

```
> bind rewrite global NOPOLICY 100 -type REQ_DEFAULT -invoke policylabel lbl-rewrt-pol
Done
> show rewrite global
1)     Global bindpoint: REQ_DEFAULT
      Number of bound policies: 1

2)     Global bindpoint: REQ_OVERRIDE
      Number of bound policies: 1
Done
```

To invoke a policy label from an Integrated Caching policy bank by using the command line interface

At the command prompt, type the following commands to invoke a policy label from an Integrated Caching policy bank and verify the configuration:

- bind cache global NOPOLICY -priority <priority> -gotoPriorityExpression <gotopriorityExpression> -type REQ_OVERRIDE | REQ_DEFAULT | RES_OVERRIDE | RES_DEFAULT -invoke reqvserver | resvserver | policylabel <policyLabelName> | <vserverName>
- show cache global

Example

```
bind cache global NOPOLICY -priority 100 -gotoPriorityExpression END -type REQ_DEFAULT -invoke policylabel lbl-cache-pol
Done
> show cache global
1)    Global bindpoint: REQ_DEFAULT
      Number of bound policies: 2

2)    Global bindpoint: RES_DEFAULT
      Number of bound policies: 1

Done
```

To invoke a policy label from a Responder policy bank by using the command line interface

At the command prompt, type the following commands to invoke a policy label from a Responder policy bank and verify the configuration:

- bind responder global NOPOLICY <priority> <gotopriorityExpression> -type OVERRIDE | DEFAULT -invoke vserver | policylabel <policyLabelName> | <vserverName>
- show responder global

Example

```
> bind responder global NOPOLICY 100 NEXT -type DEFAULT -invoke policylabel lbl-respndr-pol
Done
> show responder global
1)    Global bindpoint: REQ_DEFAULT
      Number of bound policies: 2

Done
```

To configure a policy label by using the configuration utility

1. In the navigation pane, expand the feature for which you want to configure a policy label, and then click Policy Labels. The choices are Integrated Caching, Rewrite, or Responder.
2. In the details pane, double-click the label that you want to configure.
3. If you are adding a new policy to this policy label, click Insert Policy, and in the Policy Name field, select New Policy. For more information about adding a policy, see "Creating or Modifying a Policy." Note that if you are invoking a policy bank, and do not want a rule to be evaluated prior to the invocation, click Insert Policy, and in the Policy Name field select NOPOLICY.
4. For each entry in this policy label, configure the following:
   Policy Name:
   This is already determined by the Policy Name, new policy, or NOPOLICY entry that you inserted in this bank.
   Priority:
   A numeric value that determines either an absolute order of evaluation within the bank, or is used in conjunction with a Goto expression.
   Expression:
   The policy rule. Policy expressions are described in detail in the following chapters. For an introduction, see "Configuring Default Syntax Expressions: Getting Started."
   Action:
   The action to be taken if this policy evaluates to TRUE.
   Goto Expression:
   Optional. Used to augment the Priority level to determine the next policy or policy bank to evaluate. For more information on possible values for a Goto expression, see the table "Entries in a Policy Bank."
   Invoke:
   Optional. Invokes another policy bank.
5. Click Ok. A message in the status bar indicates that the policy label is configured successfully.

Configuring a Policy Bank for a Virtual Server

Updated: 2013-09-02

You can configure a bank of policies for a virtual server. The policy bank can contain individual policies, and each entry in the policy bank can optionally invoke a policy label or a bank of policies that you configured for another virtual server. If you invoke a policy label or policy bank, you can do so without triggering an expression (a rule) by selecting a NOPOLICY "dummy" entry instead of a policy name.

To add policies to a virtual server policy bank by using the command line interface

At the command prompt, type the following commands to add policies to a virtual server policy bank and verify the configuration:

- bind lb | cs vserver <virtualServerName> <serviceType> [-policyName <policyName>] [-priority <positiveInteger>] [-gotoPriorityExpression <expression>] [-type REQUEST | RESPONSE]
- show lb | cs vserver <virtualServerName>

Example

```
add lb vserver vs-cont-sw TCP
```

```
Done
show lb vserver vs-cont-sw
    vs-cont-sw (0.0.0.0:0) - TCP    Type: ADDRESS
    State: DOWN
    Last state change was at Wed Aug 19 10:04:02 2009 (+279 ms)
    Time since last state change: 0 days, 00:02:14.420
    Effective State: DOWN
    Client Idle Timeout: 9000 sec
    Down state flush: ENABLED
    Disable Primary Vserver On Down : DISABLED
    No. of Bound Services :  0 (Total)      0 (Active)
    Configured Method: LEASTCONNECTION
    Mode: IP
    Persistence: NONE
    Connection Failover: DISABLED
Done
```

To invoke a policy label from a virtual server policy bank with a NOPOLICY entry by using the command line interface

At the command prompt, type the following commands to invoke a policy label from a virtual server policy bank with a NOPOLICY entry and verify the configuration:

- bind lb|cs vserver <virtualServerName> -policyName NOPOLICY_REWRITE|NOPOLICY_CACHE|NOPOLICY_RESPONDER -priority <integer> -type REQUEST|RESPONSE -gotoPriorityExpression <gotopriorityExpression> -invoke reqVserver|resVserver|policyLabel <vserverName>|<labelName>
- show lb vserver

Example

```
> bind lb vserver vs-cont-sw -policyname NOPOLICY-REWRITE -priority 200 -type REQUEST -gotoPriorityExpression NEXT -invoke policyLabel lbl-rewrt-pol
 Done
```

To configure a virtual server policy bank by using the configuration utility

1. In the left navigation pane, expand Traffic Management > Load Balancing, Traffic Management > Content Switching, Traffic Management > SSL Offload, Security > AAA - Application Traffic, or NetScaler Gateway, as appropriate, and then click Virtual Servers.
2. In the details pane, select the virtual server that you want to configure, and then click Open.
3. In the Configure Virtual Server dialog box click the Policies tab.
4. To create a new policy in this bank, click the icon for the type of policy or policy label that you want to add to the virtual server's bank of policies, click Insert Policy. Note that if you want to invoke a policy label without evaluating a policy rule, select the NOPOLICY "dummy" policy.
5. To configure an existing entry in this policy bank, enter the following:
   Priority:
   A numeric value that determines either an absolute order of evaluation within the bank or is used in conjunction with a Goto expression.
   Expression:
   The policy rule. Policy expressions are described in detail in the following chapters. For an introduction, see "Configuring Default Syntax Expressions: Getting Started."
   Action: on:
   The action to be taken if this policy evaluates to TRUE.
   Goto Expression:
   Optional. Determines the next policy or policy bank to evaluate. For more information on possible values for a Goto expression, see "Entries in a Policy Bank."
   Invoke:
   Optional. To invoke another policy bank, select the name of the policy label or virtual server policy bank that you want to invoke.
6. When you are done, click OK. A message in the status bar indicates that the policy is configured successfully.

Invoking or Removing a Policy Label or Virtual Server Policy Bank

Nov 14, 2013

Unlike a policy, which can only be bound once, you can use a policy label or a virtual server's policy bank any number of times by invoking it. Invocation can be performed from two places:

- From the binding for a named policy in a policy bank.
- From the binding for a NOPOLICY "dummy" entry in a policy bank.

Typically, the policy label must be of the same type as the policy from which it is invoked. For example, you would invoke a responder policy label from a responder policy.

Note: When binding or unbinding a global NOPOLICY entry in a policy bank at the command line, you specify a priority to distinguish one NOPOLICY entry from another.

To invoke a rewrite or integrated caching policy label by using the command line interface

At the command prompt, type the one of the following commands to invoke a rewrite or integrated caching policy label and verify the configuration:

- bind cache global <policy> -priority <positive_integer> [**-gotoPriorityExpression** <expression>] **-type REQ_OVERRIDE**|**REQ_DEFAULT**|**RES_OVERRIDE**|**RES_DEFAULT**] **-invoke reqvserver**|**resvserver**|**policylabel** <label_name>
- bind rewrite global<policy> -priority <positive_integer> [**-gotoPriorityExpression** <expression>] **-type REQ_OVERRIDE**|**REQ_DEFAULT**|**RES_OVERRIDE**|**RES_DEFAULT**] **-invoke reqvserver**|**resvserver**|**policylabel** <label_name>
- show cache global|show rewrite global

Example

```
> bind cache global _nonPostReq2 -priority 100 -type req_override -invoke
policylabel lbl-cache-pol
 Done
> show cache global
1)    Global bindpoint: REQ_DEFAULT
      Number of bound policies: 2

2)    Global bindpoint: RES_DEFAULT
      Number of bound policies: 1

3)    Global bindpoint: REQ_OVERRIDE
      Number of bound policies: 1

 Done
```

To invoke a responder policy label by using the command line interface

At the command prompt, type the following commands to invoke a responder policy label and verify the configuration:

- **bind responder global** <policy_Name> <priority_as_positive_integer> [<gotoPriorityExpression>] **-type REQ_OVERRIDE**|**REQ_DEFAULT**|**OVERRIDE**|**DEFAULT -invoke vserver**|**policylabel** <label_name>
- **show responder global**

Example

```
> bind responder global pol404Error1 300 -invoke policylabel lbl-respndr-pol
 Done
> show responder global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 2

 Done
>
```

To invoke a Virtual Server Policy Bank by using the command line interface

At the command prompt, type the following commands to invoke a Virtual Server Policy Bank and verify the configuration:

- **bind lb vserver** <vserver_name> **-policyName** <policy_Name> **-priority** <positive_integer> [**-gotoPriorityExpression** <expression>] **-type REQUEST**|**RESPONSE -invoke reqvserver**|**resvserver**|**policylabel** <policy_Label_Name>
- **bind lb vserver** <vserver_name>

Example

```
> bind lb vserver lbvip -policyName ns_cmp_msapp -priority 100
 Done

> show lb vserver lbvip
       lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
       State: DOWN
       Last state change was at Wed Jul 15 05:54:24 2009 (+166 ms)
       Time since last state change: 28 days, 06:37:49.250
       Effective State: DOWN
       Client Idle Timeout: 180 sec
       Down state flush: ENABLED
       Disable Primary Vserver On Down : DISABLED
       Port Rewrite : DISABLED
       No. of Bound Services :  0 (Total)      0 (Active)
       Configured Method: LEASTCONNECTION
       Mode: IP
       Persistence: NONE
       Vserver IP and Port insertion: OFF
       Push: DISABLED  Push VServer:
       Push Multi Clients: NO
       Push Label Rule: none

1)     CSPolicy: pol-cont-sw   CSVserver: vs-cont-sw   Priority: 100   Hits: 0

1)     Policy : pol-ssl Priority:0
2)     Policy : ns_cmp_msapp Priority:100
3)     Policy : cf-pol Priority:1      Inherited
 Done
```

>

To remove a rewrite or integrated caching policy label by using the command line interface

At the command prompt, type one of the following commands to remove a rewrite or integrated caching policy label and verify the configuration:

- unbind rewrite global <policyName> -priority <positiveInteger> -type **REQ_OVERRIDE**|**REQ_DEFAULT**|**RES_OVERRIDE**|**RES_DEFAULT**
- unbind cache global <policyName> -priority <positiveInteger> -type **REQ_OVERRIDE**|**REQ_DEFAULT**|**RES_OVERRIDE**|**RES_DEFAULT**
- show rewrite global|show cache global

Example

```
> unbind rewrite global NOPOLICY -priority 100 -type REQ_OVERRIDE
 Done
> show rewrite global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 1

 Done
```

To remove a responder policy label by using the command line interface

At the command prompt, type the following commands to remove a responder policy label and verify the configuration:

- unbind responder global <policyName> -priority <positiveInteger> **-type OVERRIDE**|**DEFAULT**
- **show responder global**

Example

```
> unbind responder global NOPOLICY -priority 100 -type REQ_DEFAULT
 Done
> show responder global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 1

 Done
```

To remove a Virtual Server policy label by using the command line interface

At the command prompt, type one of the following commands to remove a Virtual Server policy label and verify the configuration:

- unbind lb vserver <virtualServerName> -policyName **NOPOLICY-REWRITE**|**NOPOLICY-RESPONDER**|**NOPOLICY-CACHE -type REQUEST**|**RESPONSE -priority** <positiveInteger>
- unbind cs vserver <virtualServerName> -policyName **NOPOLICY-REWRITE**|**NOPOLICY-RESPONDER**|**NOPOLICY-CACHE -type REQUEST**|**RESPONSE -priority** <positiveInteger>
- show lb vserver|show cs vserver

Example

```
> unbind lb vserver lbvip -policyName ns_cmp_msapp -priority 200
 Done
> show lb vserver lbvip
```

```
lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
State: DOWN
Last state change was at Wed Jul 15 05:54:24 2009 (+161 ms)
Time since last state change: 28 days, 06:47:54.600
Effective State: DOWN
Client Idle Timeout: 180 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
Port Rewrite : DISABLED
No. of Bound Services :  0 (Total)      0 (Active)
Configured Method: LEASTCONNECTION
Mode: IP
Persistence: NONE
Vserver IP and Port insertion: OFF
Push: DISABLED  Push VServer:
Push Multi Clients: NO
Push Label Rule: none

1)     CSPolicy: pol-cont-sw   CSVserver: vs-cont-sw   Priority: 100   Hits: 0


1)     Policy : pol-ssl Priority:0
2)     Policy : cf-pol Priority:1      Inherited
 Done
```

To invoke a policy label or virtual server policy bank by using the configuration utility

1. Bind a policy, as described in "Binding a Policy Globally", "Binding a Policy to a Virtual Server", or "Binding a Policy to a Policy Label." Alternatively, you can enter a NOPOLICY “dummy” entry instead of a policy name. You do this if you do not want to evaluate a policy before evaluating the policy bank.
2. In the Invoke field, select the name of the policy label or virtual server policy bank that you want to evaluate if traffic matches the bound policy. A message in the status bar indicates that the policy label or virtual server policy bank is invoked successfully.

To remove a policy label invocation by using the configuration utility

1. Open the policy and clear the Invoke field. Unbinding the policy also removes the invocation of the label. A message in the status bar indicates that the policy label is removed successfully.

Configuring and Binding Policies with the Policy Manager

Sep 02, 2013

Some applications provide a specialized Policy Manager in the NetScaler configuration utility to simplify configuring policy banks. It also lets you find and delete policies and actions that are not being used.

The Policy Manager is currently available for the Rewrite, Integrated Caching, Responder, and Compression features.

The following are keyboard equivalents for the procedures in this section:

- For editing a cell in the Policy Manager, you can tab to the cell and click F2 or press the SPACE bar on the keyboard.
- To select an entry in a drop-down menu, you can tab to the entry, press the space bar to view the drop-down menu, use the UP and DOWN ARROW keys to navigate to the entry that you want, and press the space bar again to select the entry.
- To cancel a selection in a drop-down menu, press the Escape key.
- To insert a policy, tab to the row above the insertion point and press Control + Insert, or click Insert Policy.
- To remove a policy, tab to the row that contains the policy and press Delete.

Note: Note that when you delete the policy, the NetScaler searches the Goto Expression values of other policies in the bank. If any of these Goto Expression values match the priority level of the deleted policy, they are removed.

To configure policy bindings by using the Policy Manager

1. In the navigation pane, click the feature for which you want to configure policies. The choices are Responder, Integrated Caching, Rewrite or Compression.
2. In the details pane, click Policy Manager.
3. If you are configuring classic policy bindings for compression, in the Compression Policy Manager dialog box, click **Switch to Classic Syntax**. The dialog box switches to the classic syntax view and displays the Switch to Default Syntax button. At any time before you complete configuring policy bindings, if you want to configure bindings for policies that use the default syntax, click the Switch to Default Syntax button.
4. For features other than Responder, to specify the bind point, click Request or Response, and then click one of the request-time or response-time bind points. The options are Override Global, LB Virtual Server, CS Virtual Server, Default Global, or Policy Label. If you are configuring the Responder, the Request and Response flow types are not available.
5. To bind a policy to this bind point, click Insert Policy, and select a previously configured policy, a NOPOLICY label, or the New policy option. Depending on the option that you select, you have the following choices:
   - **New policy:** Create the policy as described in "Creating or Modifying a Policy," and then configure the priority level, GoTo expression, and policy invocation as described in the table, "Format of Each Entry in a Policy Bank."
   - **Existing policy**, **NOPOLICY**, or **NOPOLICY<feature name>**: Configure the priority level, GoTo expression, and policy invocation as described in the table, "Format of Each Entry in a Policy Bank." The **NOPOLICY** or **NOPOLICY<feature name>** options are available only for policies that use default syntax expressions.
6. Repeat the preceding steps to add entries to this policy bank.
7. To modify the priority level for an entry, you can do any of the following:
   - Double-click the Priority field for an entry and edit the value.
   - Click and drag a policy to another row in the table.
   - Click Regenerate Priorities.

   In all three cases, priority levels of all other policies are modified as needed to accommodate the new value. Goto Expressions with integer values are also updated automatically. For example, if you change a priority value of 10 to 100,

all policies with a Goto Expression value of 10 are updated to the value 100.

8. To change the policy, action, or policy bank invocation for an row in the table, click the down arrow to the right of the entry and do one of the following:
   - To change the policy, select another policy name or select New Policy and follow the steps in "Creating or Modifying a Policy."
   - To change the Goto Expression, select Next, End, USE_INVOCATION_RESULT, or select more and enter an expression whose result returns the priority level of another entry in this policy bank.
   - To modify an invocation, select an existing policy bank, or click New Policy Label and follow the steps in "Binding a Policy to a Policy Label."
9. To unbind a policy or a policy label invocation from this bank, click any field in the row that contains the policy or policy label, and then click Unbind Policy.
10. When you are done, click Apply Changes. A message in the status bar indicates that the policy is bound successfully.

To remove unused policies by using the Policy Manager

1. In the navigation pane, click the feature for which you want to configure the policy bank. The choices are Responder, Integrated Caching, or Rewrite.
2. In the details pane, click <Feature Name> policy manager.
3. In the <Feature Name> Policy Manager dialog box, click Cleanup Configuration.
4. In the Cleanup Configuration dialog box, select the items that you want to delete, and then click Remove.
5. In the Remove dialog box, click Yes.
6. Click Close. A message in the status bar indicates that the policy is removed successfully.

Configuring Default Syntax Expressions: Getting Started

May 25, 2015

Default syntax policies evaluate data on the basis of information that you supply in default syntax expressions. A default syntax expression analyzes data elements (for example, HTTP headers, source IP addresses, the NetScaler system time, and POST body data). In addition to configuring a default syntax expression in a policy, in some NetScaler features you configure default syntax expressions outside of the context of a policy.

To create a default syntax expression, you select a prefix that identifies a piece of data that you want to analyze, and then you specify an operation to perform on the data. For example, an operation can match a piece of data with a text string that you specify, or it can transform a text string into an HTTP header. Other operations match a returned string with a set of strings or a string pattern. You configure compound expressions by specifying Boolean and arithmetic operators, and by using parentheses to control the order of evaluation.

Default syntax expressions can also contain classic expressions. You can assign a name to a frequently used expression to avoid having to build the expression repeatedly.

Policies and a few other entities include rules that the NetScaler uses to evaluate a packet in the traffic flowing through it, to extract data from the NetScaler system itself, to send a request (a “callout”) to an external application, or to analyze another piece of data. A rule takes the form of a logical expression that is compared against traffic and ultimately returns values of TRUE or FALSE.

The elements of the rule can themselves return TRUE or FALSE, string, or numeric values.

Before configuring a default syntax expression, you need to understand the characteristics of the data that the policy or other entity is to evaluate. For example, when working with the Integrated Caching feature, a policy determines what data can be stored in the cache. With Integrated Caching, you need to know the URLs, headers, and other data in the HTTP requests and responses that the NetScaler receives. With this knowledge, you can configure policies that match the actual data and enable the NetScaler to manage caching for HTTP traffic. This information helps you determine the type of expression that you need to configure in the policy.

Basic Elements of a Default Syntax Expression

May 25, 2015

A default syntax expression consists of, at a minimum, a prefix (or a single element used in place of a prefix). Most expressions also specify an operation to be performed on the data that the prefix identifies. You format an expression of up to 1,499 characters as follows:

```
<prefix>.<operation> [<compound-operator> <prefix>.<operation>. . .]
```

where

<prefix>

is an anchor point for starting an expression.

The prefix is a period-delimited key that identifies a unit of data. For example, the following prefix examines HTTP requests for the presence of a header named Content-Type:

```
http.req.header("Content-Type")
```

Prefixes can also be used on their own to return the value of the object that the prefix identifies.

<operation>

identifies an evaluation that is to be performed on the data identified by the prefix.

For example, consider the following expression:

```
http.req.header("Content-Type").eq("text/html")
```

In this expression, the following is the operator component:

```
eq("text/html")
```

This operator causes the NetScaler to evaluate any HTTP requests that contain a Content-Type header, and in particular, to determine if the value of this header is equal to the string “text/html.” For more information, see "Operations."

<compound-operator>

is a Boolean or arithmetic operator that forms a compound expression from multiple prefix or prefix.operation elements.

For example, consider the following expression:

```
http.req.header("Content-Type").eq("text/html") && http.req.url.contains(".html")
```

This document includes the following details:

- Prefixes
- Single-Element Expressions
- Operations
- Basic Operations on Expression Prefixes

Prefixes

Updated: 2013-09-30

An expression prefix represents a discrete piece of data. For example, an expression prefix can represent an HTTP URL, an HTTP Cookie header, or a string in the body of an HTTP POST request. An expression prefix can identify and return a wide

variety of data types, including the following:

- A client IP address in a TCP/IP packet
- NetScaler system time
- An external callout over HTTP
- A TCP or UDP record type

In most cases, an expression prefix begins with one of the following keywords:

CLIENT:

Identifies a characteristic of the client that is either sending a request or receiving a response, as in the following examples:

- The prefix `client.ip.dst` designates the destination IP address in the request or response.
- The prefix `client.ip.src` designates the source IP address.

HTTP:

Identifies an element in an HTTP request or a response, as in the following examples:

- The prefix `http.req.body(integer)` designates the body of the HTTP request as a multiline text object, up to the character position designated in integer.
- The prefix `http.req.header("header_name")` designates an HTTP header, as specified in header_name.
- The prefix `http.req.url` designates an HTTP URL in URL-encoded format.

SERVER:

Identifies an element in the server that is either processing a request or sending a response.

SYS:

Identifies a characteristic of the NetScaler that is processing the traffic.

Note: Note that DNS policies support only SYS, CLIENT, and SERVER objects.

In addition, in the NetScaler Gateway, the Clientless VPN function can use the following types of prefixes:

TEXT:

Identifies any text element in a request or a response.

TARGET:

Identifies the target of a connection.

URL:

Identifies an element in the URL portion of an HTTP request or response.

As a general rule of thumb, any expression prefix can be a self-contained expression. For example, the following prefix is a complete expression that returns the contents of the HTTP header specified in the string argument (enclosed in quotation marks):

```
http.res.header.("myheader")
```

Or you can combine prefixes with simple operations to determine TRUE and FALSE values. For example, the following returns a value of TRUE or FALSE:

```
http.res.header.("myheader").exists
```

You can also use complex operations on individual prefixes and multiple prefixes within an expression, as in the following example:

```
http.req.url.length + http.req.cookie.length <= 500
```

Which expression prefixes you can specify depends on the NetScaler feature. The following table describes the expression prefixes that are of interest on a per-feature basis

Table 1. Permitted Types of Expression Prefixes in Various NetScaler Features

Feature	Types of Expression Prefix Used in the Feature
DNS	SYS, CLIENT, SERVER
Responder in Protection Features	HTTP, SYS, CLIENT
Content Switching	HTTP, SYS, CLIENT
Rewrite	HTTP, SYS, CLIENT, SERVER, URL, TEXT, TARGET, VPN
Integrated Caching	HTTP, SYS, CLIENT, SERVER
NetScaler Gateway, Clientless Access	HTTP, SYS, CLIENT, SERVER, URL, TEXT, TARGET, VPN

Note: For details on the permitted expression prefixes in a feature, see the documentation for that feature.

Single-Element Expressions

The simplest type of default syntax expression contains a single element. This element can be one of the following:

- `true`. A default syntax expression can consist simply of the value `true`. This type of expression always returns a value of TRUE. It is useful for chaining policy actions and triggering Goto expressions.
- `false`. A default syntax expression can consist simply of the value `false`. This type of expression always returns a value of FALSE.
- A prefix for a compound expression. For example, the prefix `HTTP.REQ.HOSTNAME` is a complete expression that returns a host name and `HTTP.REQ.URL` is a complete expression that returns a URL. The prefix could also be used in conjunction with operations and additional prefixes to form a compound expression.

Operations

In most expressions, you also specify an operation on the data that the prefix identifies. For example, suppose that you specify the following prefix:

`http.req.url`

This prefix extracts URLs in HTTP requests. This expression prefix does not require any operators to be used in an expression. However, when you configure an expression that processes HTTP request URLs, you can specify operations that analyze particular characteristics of the URL. Following are a few possibilities:

- Search for a particular host name in the URL.
- Search for a particular path in the URL.
- Evaluate the length of the URL.
- Search for a string in the URL that indicates a time stamp and convert it to GMT.

The following is an example of a prefix that identifies an HTTP header named Server and an operation that searches for the string IIS in the header value:

```
http.res.header("Server").contains("IIS")
```

Following is an example of a prefix that identifies host names and an operation that searches for the string “www.mycompany.com” as the value of the name:

```
http.req.hostname.eq("www.mycompany.com")
```

Basic Operations on Expression Prefixes

The following table describes a few of the basic operations that can be performed on expression prefixes.

Table 2. Basic Operations for Expressions

Operation	Determines Whether or Not
CONTAINS(<string>)	The object matches <string>. Following is an example: `http.req.header("Cache-Control").contains("no-cache")`
EXISTS	A particular item is present in an object. Following is an example: `http.res.header("MyHdr").exists`
EQ(<text>)	A particular non-numeric value is present in an object. Following is an example: `http.req.method.eq(post)`
EQ(<integer>)	A particular numeric value is present in an object. Following is an example: `client.ip.dst.eq(10.100.10.100)`
LT(<integer>)	An object's value is less than a particular value. Following is an example: `http.req.content_length.lt(5000)`
GT(<integer>)	An object's value is greater than a particular value. Following is an example: `http.req.content_length.gt(5)`

The following table summarizes a few of the available types of operations.

Table 3. Basic Types of Operations

Operation Type	Description
Text operations	Match individual strings and sets of strings with any portion of a target. The target can be an entire string, the start of a string, or any portion of text in between the start and the end of the string. For example, you can extract the string "XYZ" from "XYZSomeText". Or, you can compare an HTTP header value with an array of different strings. You can also transform text into another type of data. Following are examples: • Transform a string into an integer value • Create a list from the query strings in a URL • Transform a string into a time value
Numeric operations	Numeric operations include applying arithmetic operators, evaluating content length, the number of items in a list, dates, times, and IP addresses.

Compound Default Syntax Expressions

May 25, 2015

You can configure a default syntax expression that contains Boolean or arithmetic operators and multiple atomic operations. The following compound expression contains a boolean AND:

```
http.req.hostname.eq("mycompany.com") && http.req.method.eq(post)
```

The following expression adds the value of two targets, and compares the result to a third value:

```
http.req.url.length + http.req.cookie.length <= 500
```

A compound expression can contain any number of logical and arithmetic operators. The following expression evaluates the length of an HTTP request on the basis of its URL and cookie, evaluates text in the header, and performs a Boolean AND on these two results:

```
http.req.url.length + http.req.cookie.length <= 500 && http.req.header.contains("some text")
```

You can use parentheses to control the order of evaluation in a compound expression.

This document includes the following details:

- Booleans in Compound Expressions
- Parentheses in Compound Expressions
- Compound Operations for Strings
- Compound Operations for Numbers

Booleans in Compound Expressions

You configure compound expressions with the following operators:

&&.

This operator is a logical AND. For the expression to evaluate to TRUE, all components that are joined by the And must evaluate to TRUE. Following is an example:

```
http.req.url.hostname.eq("myHost") && http.req.header("myHeader").exists
```

||.

This operator is a logical OR. If any component of the expression that is joined by the OR evaluates to TRUE, the entire expression is TRUE.

!.

Performs a logical NOT on the expression.

In some cases, the NetScaler configuration utility offers AND, NOT, and OR operators in the Add Expression dialog box. However, these are of limited use. Citrix recommends that you use the operators &&, ||, and ! to configure compound expressions that use Boolean logic.

Parentheses in Compound Expressions

You can use parentheses to control the order of evaluation of an expression. The following is an example:

```
http.req.url.contains("myCompany.com") || (http.req.url.hostname.eq("myHost") &&
http.req.header("myHeader").exists)
```

The following is another example:

```
(http.req.header("Content-Type").exists && http.req.header("Content-Type").eq("text/html")) ||
(http.req.header("Transfer-Encoding").exists || http.req.header("Content-Length").exists)
```

Compound Operations for Strings

The following table describes operators that you can use to configure compound operations on string data.

Table 1. String-Based Operations for Compound Default Syntax Expressions

All string operations	
Operations that produce a string value	
str + str	Concatenates the value of the expression on the left of the operator with the value on the right. Following is an example: `http.req.hostname + http.req.url.protocol`
str + num	Concatenates the value of the expression on the left of the operator with a numeric value on the right. Following is an example: `http.req.hostname + http.req.url.content_length`
num + str	Concatenates the numeric value of the expression on the left side of the operator with a string value on the right. Following is an example: `http.req.url.content_length + http.req.url.hostname`
str + ip	Concatenates the string value of the expression on the left side of the operator with an IP address value on the right. Following is an example: `http.req.hostname + 10.00.000.00`
ip + str	Concatenates the IP address value of the expression on the left of the operator with a string value on the right.Following is an example: `client.ip.dst + http.req.url.hostname`
str1 ALT str2	Uses the string1 or string2 value that is derived from the expression on either side of the operator, as long as neither of these expressions is a compound expressions. Following is an example: `http.req.hostname alt client.ip.src`
Operations on strings that produce a result of TRUE or FALSE	

All string operations	
str == str	Evaluates whether the strings on either side of the operator are the same. Following is an example: http.req.header("myheader") == http.res.header("myheader")
str <= str	Evaluates whether the string on the left side of the operator is the same as the string on the right, or precedes it alphabetically.
str >= str	Evaluates whether the string on the left side of the operator is the same as the string on the right, or follows it alphabetically.
str < str	Evaluates whether the string on the left side of the operator precedes the string on the right alphabetically.
str > str	Evaluates whether the string on the left side of the operator follows the string on the right alphabetically.
str != str	Evaluates whether the strings on either side of the operator are different.
Logical operations on strings	
bool && bool	This operator is a logical AND. When evaluating the components of the compound expression, all components that are joined by the AND must evaluate to TRUE. Following is an example: http.req.method.eq(GET) && http.req.url.query.contains("viewReport && my_pagelabel")
bool || bool	This operator is a logical OR. When evaluating the components of the compound expression, if any component of the expression that is joined by the OR evaluates to TRUE, the entire expression is TRUE. Following is an example: http.req.url.contains(".js") || http.res.header.("Content-Type").contains("javascript")
!bool	Performs a logical NOT on the expression.

Compound Operations for Numbers

Updated: 2013-09-02

You can configure compound numeric expressions. For example, the following expression returns a numeric value that is the sum of an HTTP header length and a URL length:

```
http.req.header.length + http.req.url.length
```

The following tables describes operators that you can use to configure compound expressions for numeric data.

Table 2. Arithmetic Operations on Numbers

Operator	Description
num + num	Add the value of the expression on the left of the operator to the value of the expression on the right. Following is an example: `http.req.content_length + http.req.url.length`
num – num	Subtract the value of the expression on the right of the operator from the value of the expression on the left.
num * num	Multiply the value of the expression on the left of the operator with the value of the expression on the right. Following is an example: `client.interface.rxthroughput * 9`
num / num	Divide the value of the expression on the left of the operator by the value of the expression on the right.
num % num	Calculate the modulo, or the numeric remainder on a division of the value of the expression on the left of the operator by the value of the expression on the right. For example, the values "15 mod 4" equals 3, and "12 mod 4" equals 0.
~number	Returns a number after applying a bitwise logical negation of the number. The following example assumes that numeric.expression returns 12 (binary 1100): `~numeric.expression.` The result of applying the ~ operator is -11 (a binary 1110011, 32 bits total with all ones to the left). Note that all returned values of less than 32 bits before applying the operator implicitly have zeros to the left to make them 32 bits wide.
number ^ number	Compares two bit patterns of equal length and performs an XOR operation on each pair of corresponding bits in each number argument, returning 1 if the bits are different, and 0 if they are the same. Returns a number after applying a bitwise XOR to the integer argument and the current number value. If the values in the bitwise comparison are the same, the returned value is a 0. The following example assumes that numeric.expression1 returns 12 (binary 1100) and numeric.expression2 returns 10 (binary 1010): `numeric.expression1 ^ numeric.expression2`

Operator	Description
	The result of applying the ^ operator to the entire expression is 6 (binary 0110). Note that all returned values of less than 32 bits before applying the operator implicitly have zeros to the left to make them 32 bits wide.
number | number	Returns a number after applying a bitwise OR to the number values. If either value in the bitwise comparison is a 1, the returned value is a 1. The following example assumes that numeric.expression1 returns 12 (binary 1100) and numeric.expression2 returns 10 (binary 1010): `numeric.expression1 | numeric.expression2` The result of applying the | operator to the entire expression is 14 (binary 1110). Note that all returned values of less than 32 bits before applying the operator implicitly have zeros to the left to make them 32 bits wide.
number & number	Compares two bit patterns of equal length and performs a bitwise AND operation on each pair of corresponding bits, returning 1 if both of the bits contains a value of 1, and 0 if either bits are 0. The following example assumes that numeric.expression1 returns 12 (binary 1100) and numeric.expression2 returns 10 (binary 1010): `numeric.expression1 & numeric.expression2` The whole expression evaluates to 8 (binary 1000). Note that all returned values of less than 32 bits before applying the operator implicitly have zeros to the left to make them 32 bits wide.
num << num	Returns a number after a bitwise left shift of the number value by the right-side number argument number of bits. Note that the number of bits shifted is integer modulo 32. The following example assumes that numeric.expression1 returns 12 (binary 1100) and numeric.expression2 returns 3: `numeric.expression1 << numeric.expression2` The result of applying the LSHIFT operator is 96 (a binary 1100000). Note that all returned values of less than 32 bits before applying the operator implicitly have zeros to the left to make them 32 bits wide.
num >> num	Returns a number after a bitwise right shift of the number value by the integer argument number of bits. Note that the number of bits shifted is integer modulo 32. The following example assumes that numeric.expression1 returns 12 (binary 1100) and numeric.expression2 returns 3: `numeric.expression1 >> numeric.expression2` The result of applying the RSHIFT operator is 1 (a binary 0001).

Operator	Description
	Note that all returned values of less than 32 bits before applying the operator implicitly have zeros to the left to make them 32 bits wide.

Table 3. Numeric Operators That Produce a Result of TRUE or FALSE

Operator	Description
num == num	Determine if the value of the expression on the left of the operator is equal to the value of the expression on the right.
num != num	Determine if the value of the expression on the left of the operator is not equal to the value of the expression on the right.
num > num	Determine if the value of the expression on the left of the operator is greater than the value of the expression on the right.
num < num	Determine if the value of the expression on the left of the operator is less than the value of the expression on the right.
num >= num	Determine if the value of the expression on the left of the operator is greater than or equal to the value of the expression on the right.
num <= num	Determine if the value of the expression on the left of the operator is less than or equal to the value of the expression on the right

Functions for Data Types in the Policy Infrastructure

The NetScaler policy infrastructure supports the following numeric data types:

- Integer (32 bits)
- Unsigned long (64 bits)
- Double (64 bits)

Simple expressions can return all of these data types. Therefore, you can create compound expressions that use arithmetic operators and logical operators to evaluate or return values of these data types. Additionally, you can use all of these values in policy expressions. Literal constants of type unsigned long can be specified by appending the string ul to the number. Literal constants of type double contain a period (.), an exponent, or both.

Arithmetic Operators, Logical Operators, and Type Promotion

In compound expressions, the following standard arithmetic and logical operators can be used for the double and unsigned long data types:

- +, -, *, and /

- %, ~, ^, &, |, <<, and >> (do not apply to double)
- ==, !=, >, <, >=, and <=

All of these operators have the same meaning as in the C programming language.

In all cases of mixed operations between operands of type integer, unsigned long, and double, type promotion is performed so that the operation can be performed on operands of the same type. A type of lower precedence is automatically promoted to the type of the operand with the highest precedence involved in the operation. The order of precedence (higher to lower) is as follows:

- Double
- Unsigned long
- Integer

Therefore, an operation that returns a numeric result returns a result of the highest type involved in the operation.

For example, if the operands are of type integer and unsigned long, the integer operand is automatically converted to type unsigned long. This type conversion is performed even in simple expressions in which the type of data identified by the expression prefix does not match the type of data that is passed as the argument to the function. To illustrate such an example, in the operation HTTP.REQ.CONTENT_LENGTH.DIV(3ul), the integer returned by the prefix HTTP.REQ.CONTENT_LENGTH is automatically converted to unsigned long (the type of the data passed as the argument to the DIV() function), and an unsigned long division is performed. Similarly, the argument can be promoted in an expression. For example, HTTP.REQ.HEADER("myHeader").TYPECAST_DOUBLE_AT.DIV(5) promotes the integer 5 to type double and performs double-precision division.

The following table describes the arithmetic and Boolean functions that can be used with the integer, unsigned long, and double data types. For information about expressions for casting data of one type to data of another type, see "Typecasting Data."

Specifying the Character Set in Expressions

Jul 11, 2013

The policy infrastructure on the Citrix® NetScaler® appliance supports the ASCII and UTF-8 character sets. The default character set is ASCII. If the traffic for which you are configuring an expression consists of only ASCII characters, you need not specify the character set in the expression. However, you must specify the character set in every simple expression that is meant for UTF-8 traffic. To specify the UTF-8 character set in a simple expression, you must include the SET_CHAR_SET(<charset>) function, with <charset> specified as UTF_8, as shown in the following examples:

```
HTTP.REQ.BODY(10).SET_CHAR_SET(UTF_8).CONTAINS("ß")
```

```
HTTP.RES.BODY(100).SET_CHAR_SET(UTF_8).BEFORE_STR("Bücher").AFTER_STR("Wörterbuch")
```

In an expression, the SET_CHAR_SET() function must be introduced at the point in the expression after which data processing must be carried out in the specified character set. For example, in the expression HTTP.REQ.BODY(1000).AFTER_REGEX(re/following example/).BEFORE_REGEX(re/In the preceding example/).CONTAINS_ANY("Greek_ alphabet"), if the strings stored in the pattern set "Greek_alphabet" are in UTF-8, you must include the SET_CHAR_SET(UTF_8) function immediately before the CONTAINS_ANY("<string>") function, as follows:

```
HTTP.REQ.BODY(1000).AFTER_REGEX(re/following example/).BEFORE_REGEX(re/In the preceding
example/).SET_CHAR_SET(UTF_8).CONTAINS_ANY("Greek_ alphabet")
```

The SET_CHAR_SET() function sets the character set for all further processing (that is, for all subsequent functions) in the expression unless it is overridden later in the expression by another SET_CHAR_SET() function that changes the character set. Therefore, if all the functions in a given simple expression are intended for UTF-8, you can include the SET_CHAR_SET(UTF_8) function immediately after functions that identify text (for example, the HEADER("<name>") or BODY(<int>) functions). In the second example that follows the first paragraph above, if the ASCII arguments passed to the AFTER_REGEX() and BEFORE_REGEX() functions are changed to UTF-8 strings, you can include the SET_CHAR_SET(UTF_8) function immediately after the BODY(1000) function, as follows:

```
HTTP.REQ.BODY(1000).SET_CHAR_SET(UTF_8).AFTER_REGEX(re/Bücher/).BEFORE_REGEX(re/Wörterbuch/).CONTAINS_ANY("Greek_alphabet")
```

The UTF-8 character set is a superset of the ASCII character set, so expressions configured for the ASCII character set continue to work as expected if you change the character set to UTF-8.

Compound Expressions with Different Character Sets

In a compound expression, if one subset of expressions is configured to work with data in the ASCII character set and the rest of the expressions are configured to work with data in the UTF-8 character set, the character set specified for each individual expression is considered when the expressions are evaluated individually. However, when processing the compound expression, just before processing the operators, the appliance promotes the character set of the returned ASCII values to UTF-8. For example, in the following compound expression, the first simple expression evaluates data in the ASCII character set while the second simple expression evaluates data in the UTF-8 character set:

```
HTTP.REQ.HEADER("MyHeader") == HTTP.REQ.BODY(10).SET_CHAR_SET(UTF_8)
```

However, when processing the compound expression, just before evaluating the "is equal to" Boolean operator, the NetScaler appliance promotes the character set of the value returned by HTTP.REQ.HEADER("MyHeader") to UTF-8.

The first simple expression in the following example evaluates data in the ASCII character set. However, when the NetScaler appliance processes the compound expression, just before concatenating the results of the two simple expressions, the appliance promotes the character set of the value returned by HTTP.REQ.BODY(10) to UTF-8.

```
HTTP.REQ.BODY(10) + HTTP.REQ.HEADER("MyHeader").SET_CHAR_SET(UTF_8)
```

Consequently, the compound expression returns data in the UTF-8 character set.

Specifying the Character Set Based on the Character Set of Traffic

You can set the character set to UTF-8 on the basis of traffic characteristics. If you are not sure whether the character set of the traffic being evaluated is UTF-8, you can configure a compound expression in which the first expression checks for UTF-8 traffic and subsequent expressions set the character set to UTF-8. Following is an example of a compound expression that first checks the value of "charset" in the request's Content-Type header for "UTF-8" before checking whether the first 1000 bytes in the request contain the UTF-8 string Bücher:

```
HTTP.REQ.HEADER("Content-Type").SET_TEXT_MODE(IGNORECASE).TYPECAST_NVLIST_T('=', '; ', '"').VALUE("charset").EQ("UTF-8") &&
HTTP.REQ.BODY(1000).SET_CHAR_SET(UTF_8).CONTAINS("Bücher")
```

If you are sure that the character set of the traffic being evaluated is UTF-8, the second expression in the example is sufficient.

Character and String Literals in Expressions

During expression evaluation, even if the current character set is ASCII, character literals and string literals, which are enclosed in single quotation marks (') and quotation marks (""), respectively, are considered to be literals in the UTF-8 character set. In a given expression, if a function is operating on character or string literals in the ASCII character set and you include a non-ASCII character in the literal, an error is returned.

Values in Hexadecimal and Octal Formats

When configuring an expression, you can enter values in octal and hexadecimal formats. However, each hexadecimal or octal byte is considered a UTF-8 byte. Invalid UTF-8 bytes result in errors regardless of whether the value is entered manually or pasted from the clipboard. For example, "\xce\x20" is an invalid UTF-8 character because "c8" cannot be followed by "20" (each byte in a multi-byte UTF-8 string must have the high bit set). Another example of an invalid UTF-8 character is "\xce \xa9," since the hexadecimal characters are separated by a white-space character.

Functions That Return UTF-8 Strings

Only the <text>.XPATH and <text>.XPATH_JSON functions always return UTF-8 strings. The following MYSQL routines determine at runtime which character set to return, depending on the data in the protocol:

- MYSQL_CLIENT_T.USER
- MYSQL_CLIENT_T.DATABASE
- MYSQL_REQ_QUERY_T.COMMAND
- MYSQL_REQ_QUERY_T.TEXT
- MYSQL_REQ_QUERY_T.TEXT(<unsigned int>)
- MYSQL_RES_ERROR_T.SQLSTATE
- MYSQL_RES_ERROR_T.MESSAGE
- MYSQL_RES_FIELD_T.CATALOG
- MYSQL_RES_FIELD_T.DB
- MYSQL_RES_FIELD_T.TABLE
- MYSQL_RES_FIELD_T.ORIGINAL_TABLE
- MYSQL_RES_FIELD_T.NAME
- MYSQL_RES_FIELD_T.ORIGINAL_NAME
- MYSQL_RES_OK_T.MESSAGE
- MYSQL_RES_ROW_T.TEXT_ELEM(<unsigned int>)

Terminal Connection Settings for UTF-8

When you set up a connection to the NetScaler appliance by using a terminal connection (by using PuTTY, for example), you must set the character set for transmission of data to UTF-8.

Classic Expressions in Default Syntax Expressions

Mar 20, 2012

Classic expressions describe basic characteristics of traffic. In some cases, you may want to use a classic expression in a default syntax expression. You can do so with the default syntax expression configuration tool. This can be helpful when manually migrating the older classic expressions to the default syntax.

Note that when you upgrade the NetScaler to version 9.0 or higher, Integrated Caching policies are automatically upgraded to default syntax policies, and the expressions in these policies are upgraded to the default syntax.

The following is the syntax for all default syntax expressions that use a classic expression:

```
SYS.EVAL_CLASSIC_EXPR("expression")
```

Following are examples of the SYS.EVAL_CLASSIC_EXPR("expression") expression:

```
sys.eval_classic_expr("req.ssl.client.cipher.bits > 1000")
sys.eval_classic_expr("url contains abc")
sys.eval_classic_expr("req.ip.sourceip == 10.102.1.61 -netmask 255.255.255.255")
sys.eval_classic_expr("time >= *:30:00GMT")
sys.eval_classic_expr("e1 || e2")
sys.eval_classic_expr("req.http.urllen > 50")
sys.eval_classic_expr("dayofweek == wedGMT")
```

Configuring Default Syntax Expressions in a Policy

Sep 02, 2013

You can configure a default syntax expression of up to 1,499 characters in a policy. The user interface for default syntax expressions depends to some extent on the feature for which you are configuring the expression, and on whether you are configuring an expression for a policy or for another use.

When configuring expressions on the command line, you delimit the expression by using quotation marks (“. . .” or '. . .'). Within an expression, you escape additional quotation marks by using a back-slash (\). For example, the following are standard methods for escaping quotation marks in an expression:

"\"abc\""

‘\"abc\"’

You must also use a backslash to escape question marks and other backslashes on the command line. For example, the expression http.req.url.contains(“\?”) requires a backslash so that the question mark is parsed. Note that the backslash character will not appear on the command line after you type the question mark. On the other hand, if you escape a backslash (for example, in the expression 'http.req.url.contains("\\\\http")'), the escape characters are echoed on the command line.

To make an entry more readable, you can escape the quotation marks for an entire expression. At the start of the expression you enter the escape sequence “q” plus one of the following special characters: /{<|~$^+=&%@`?.

You enter only the special character at the end of the expression, as follows:

q@http.req.url.contains("sometext") && http.req.cookie.exists@

q~http.req.url.contains("sometext") && http.req.cookie.exists~

Note that an expression that uses the { delimiter is closed with }.

For some features (for example, Integrated Caching and Responder), the policy configuration dialog box provides a secondary dialog box for configuring expressions. This dialog enables you to choose from drop-down lists that show the available choices at each point during expression configuration. You cannot use arithmetic operators when using these configuration dialogs, but most other default syntax expression features are available. To use arithmetic operators , write your expressions in free-form format.

To configure a default syntax rule by using the command line interface

At the command prompt, type the following commands to configure a default syntax rule and verify the configuration:

1. add cache|dns|rewrite|cs policy policyName **-rule** expression featureSpecificParameters **-action**
2. show cache|dns|rewrite|cs policy policyName

   Following is an example of configuring a caching policy:

   Example

```
> add cache policy pol-cache -rule http.req.content_length.le(5) -action INVAL
 Done

> show cache policy pol-cache
```

```
    Name: pol-cache
    Rule: http.req.content_length.le(5)
    CacheAction: INVAL
    Invalidate groups: DEFAULT
    UndefAction: Use Global
    Hits: 0
    Undef Hits: 0

Done
```

To configure a default syntax policy expression by using the configuration utility

1. In the navigation pane, click the name of the feature where you want to configure a policy, for example, you can select Integrated Caching, Responder, DNS, Rewrite, or Content Switching, and then click Policies.
2. Click Add.
3. For most features, click in the Expression field. For Content Switching, click Configure.
4. Click the Prefix icon (the house) and select the first expression prefix from the drop-down list. For example, in Responder, the options are HTTP, SYS, and CLIENT. The next set of applicable options appear in a drop-down list.
5. Double-click the next option to select it, and then type a period (.). Again, a set of applicable options appears in another drop-down list.
6. Continue selecting options until an entry field (signalled by parentheses) appears. When you see an entry field, enter an appropriate value in the parentheses. For example, if you select GT(int) (greater-than, integer format), you specify an integer in the parentheses. Text strings are delimited by quotation marks. Following is an example:
   `HTTP.REQ.BODY(1000).BETWEEN("this","that")`
7. To insert an operator between two parts of a compound expression, click the Operators icon (the sigma), and select the operator type. Following is an example of a configured expression with a Boolean OR (signalled by double vertical bars, ||):
   `HTTP.REQ.URL.EQ("www.mycompany.com")||HTTP.REQ.BODY(1000).BETWEEN("this","that")`
8. To insert a named expression, click the down arrow next to the Add icon (the plus sign) and select a named expression.
9. To configure an expression using drop-down menus, and to insert built-in expressions, click the Add icon (the plus sign). The Add Expression dialog box works in a similar way to the main dialog box, but it provides drop-down lists for selecting options, and it provides text fields for data entry instead of parentheses. This dialog box also provides a Frequently Used Expressions drop-down list that inserts commonly used expressions. When you are done adding the expression, click OK.
10. When finished, click Create. A message in the status bar indicates that the policy expression is configured successfully.

To test a default syntax expression by using the configuration utility

1. In the navigation pane, click the name of the feature for which you want to configure a policy (for example, you can select Integrated Caching, Responder, DNS, Rewrite, or Content Switching), and then click Policies.
2. Select a policy and click Open.
3. To test the expression, click the Evaluate icon (the check mark).
4. In the expression evaluator dialog box, select the Flow Type that matches the expression.
5. In the HTTP Request Data or HTTP Response Data field, paste the HTTP request or response that you want to parse with the expression, and click Evaluate. Note that you must supply a complete HTTP request or response, and the header and body should be separated by blank line. Some programs that trap HTTP headers do not also trap the response. If you are copying and pasting only the header, insert a blank line at the end of the header to form a complete HTTP request or response.

6. Click Close to close this dialog box.

Configuring Named Default Syntax Expressions

Oct 29, 2013

Instead of retyping the same expression multiple times in multiple policies, you can configure a named expression and refer to the name any time you want to use the expression in a policy. For example, you could create the following named expressions:

ThisExpression:

```
http.req.body(100).contains("this")
```

ThatExpression:

```
http.req.body(100).contains("that")
```

You can then use these named expressions in a policy expression. For example, the following is a legal expression based on the preceding examples:

```
ThisExpression || ThatExpression
```

You can use the name of a default syntax expression as the prefix to a function. The named expression can be either a simple expression or a compound expression. The function must be one that can operate on the type of data that is returned by the named expression.

Example 1: Simple Named Expression as a Prefix

The following simple named expression, which identifies a text string, can be used as a prefix to the AFTER_STR("<string>")function, which works with text data:

```
HTTP.REQ.BODY(1000)
```

If the name of the expression is top1KB, you can use top1KB.AFTER_STR("username") instead of HTTP.REQ.BODY(1000).AFTER_STR("username").

Example 2: Compound Named Expression as a Prefix

You can create a compound named expression called basic_header_value to concatenate the user name in a request, a colon (:), and the user's password, as follows:

```
add policy expression basic_header_value "HTTP.REQ.USER.NAME + \":\" + HTTP.REQ.USER.PASSWD"
```

You can then use the name of the expression in a rewrite action, as shown in the following example:

```
add rewrite action insert_b64encoded_authorization insert_http_header authorization '"Basic " + basic_header_value.b64encode' -bypassSafetyCheck YES
```

In the example, in the expression that is used to construct the value of the custom header, the B64 encoding algorithm is applied to the string returned by the compound named expression.

You can also use a named expression (either by itself or as a prefix to a function) to create the text expression for the replacement target in a rewrite.

To configure a named default syntax expression by using the command line interface

At the command prompt, type the following commands to configure a named expression and verify the configuration:

- add policy expression <name><value>

- show policy expression <name>
  Example

  ```
  > add policy expression myExp "http.req.body(100).contains(\"the other\")"
   Done

  > show policy expression myExp
  1)     Name: myExp  Expr: "http.req.body(100).contains("the other")"  Hits: 0 Type : ADVANCED
   Done
  ```

  The expression can be up to 1,499 characters.

To configure a named expression by using the configuration utility

1. In the navigation pane, expand AppExpert, and then click Expressions.
2. Click Advanced Expressions.
3. Click Add.
4. Enter a name and a description for the expression.
5. Configure the expression by using the process described in "To configure a default syntax policy expression by using the configuration utility." A message in the status bar indicates that the policy expression is configured successfully.

Configuring Default Syntax Expressions Outside the Context of a Policy

Oct 29, 2013

A number of functions, including the following, can require a default syntax expression that is not part of a policy:

Integrated Caching selectors:

You define multiple non-compound expressions (selectlets) in the definition of the selector. Each selectlet is in an implicit logical AND relationship with the others.

Load Balancing:

You configure an expression for the TOKEN method of load balancing for a load balancing virtual server.

Rewrite actions:

Expressions define the location of the rewrite action and the type of rewriting to be performed, depending on the type of rewrite action that you are configuring. For example, a DELETE action only uses a target expression. A REPLACE action uses a target expression and an expression to configure the replacement text.

Rate-based policies:

You use default syntax expressions to configure Limit Selectors. You can use these selectors when configuring policies to throttle the rate of traffic to various servers. You define up to five non-compound expressions (selectlets) in the definition of the selector. Each selectlet is in an implicit logical AND with the others.

To configure a default syntax expression outside a policy by using the command line interface (cache selector example)

At the command prompt, type the following commands to configure a default syntax expression outside a policy and verify the configuration:

- add cache selector <selectorName> <rule>
- show cache selector <selectorName>

Example

```
> add cache selector mainpageSelector "http.req.cookie.value(\"ABC_def\")"
"http.req.url.query.value(\"_ghi\")"selector "mainpageSelector" added
 Done
> show cache selector mainpageSelector
      Name: mainpageSelector
      Expressions:
            1) http.req.cookie.value("ABC_def")
            2) http.req.url.query.value("_ghi")
 Done
```

Following is an equivalent command that uses the more readable q delimiter, as described in "Configuring Default Syntax Expressions in a Policy":

```
> add cache selector mainpageSelector2 q~http.req.cookie.value("ABC_def")~
q~http.req.url.query.value("_ghi")~selector "mainpageSelector2" added
 Done
> show cache selector mainpageSelector2
      Name: mainpageSelector2
      Expressions:
```

```
        1) http.req.cookie.value("ABC_def")
        2) http.req.url.query.value("_ghi")
Done
```

Default Syntax Expressions: Evaluating Text

Sep 02, 2013

You can configure a policy with a default syntax expression that evaluates text in a request or response. Default syntax text expressions can range from simple expressions that perform string matching in HTTP headers to complex expressions that encode and decode text. You can configure text expressions to be case sensitive or case insensitive and to use or ignore spaces. You can also configure complex text expressions by combining text expressions with Boolean operators

You can use expression prefixes and operators for evaluating HTTP requests, HTTP responses, and VPN and Clientless VPN data. However, text expression prefixes are not restricted to evaluating these elements of your traffic. For information about additional default syntax text expression prefixes and operators, see the following topics:

- "Pattern Sets"
- "Regular Expressions"
- "Typecasting Data"
- "Default Syntax Expressions: Parsing HTTP, TCP, and UDP Data"
- "Default Syntax Expressions: Parsing SSL Certificates"
- "Expressions for SSL Certificate Dates"

About Text Expressions

Sep 02, 2013

You can configure various expressions for working with text that flows through the NetScaler appliance. Following are some examples of how you can parse text by using a default syntax expression:

- Determine that a particular HTTP header exists.
  For example, you may want to identify HTTP requests that contains a particular Accept-Language header for the purpose of directing the request to a particular server.
- Determine that a particular HTTP URL contains a particular string.
  For example, you may want to block requests for particular URLs. Note that the string can occur at the beginning, middle, or end of another string.
- Identify a POST request that is directed to a particular application.
  For example, you may want to identify all POST requests that are directed to a database application for the purpose of refreshing cached application data.

Note that there are specialized tools for viewing the data stream for HTTP requests and responses. For example, from the following URL, you can download a Firefox Web browser plug-in that displays HTTP request and response headers:

"https://addons.mozilla.org/en-US/firefox/addon/3829"

The following plug-in displays headers, query strings, POST data, and other information:

"https://addons.mozilla.org/en-US/firefox/addon/6647"

After you download these plug-ins, they are accessible from the Firefox Tools menu.

About Operations on Text

A text-based expression consists of at least one prefix to identify an element of data and usually (although not always) an operation on that prefix. Text-based operations can apply to any part of a request or a response. Basic operations on text include various types of string matches.

For example, the following expression compares a header value with a string:

```
http.req.header("myHeader").contains("some-text")
```

Following expressions are examples of matching a file type in a request:

```
http.req.url.suffix.contains("jpeg")
```

```
http.req.url.suffix.eq("jpeg")
```

In the preceding examples, the contains operator permits a partial match and the eq operator looks for an exact match.

Other operations are available to format the string before evaluating it. For example, you can use text operations to strip out quotes and white spaces, to convert the string to all lowercase, or to concatenate strings.

Note: Complex operations are available to perform matching based on patterns or to convert one type of text format to another type.

For more information, see the following topics:

- "Pattern Sets and Data Sets."
- "Regular Expressions."
- "Typecasting Data."

Compounding and Precedence in Text Expressions

You can apply various operators to combine text prefixes or expressions. For example, the following expression concatenates the returned values of each prefix:

```
http.req.hostname + http.req.url
```

Following is an example of a compound text expression that uses a logical AND. Both components of this expression must be TRUE for a request to match the expression:

```
http.req.method.eq(post) && http.req.body(1024).startswith("destination=")
```

Note: For more information on operators for compounding, see "Compound Default Syntax Expressions."

Categories of Text Expressions

The primary categories of text expressions that you can configure are:

- Information in HTTP headers, HTTP URLs, and the POST body in HTTP requests.
  For more information, see "Expression Prefixes for Text in HTTP Requests and Responses."
- Information regarding a VPN or a clientless VPN.
  For more information, see "Expression Prefixes for VPNs and Clientless VPNs."
- TCP payload information.
  For more information about TCP payload expressions, see "Default Syntax Expressions: Parsing HTTP, TCP, and UDP Data."
- Text in a Secure Sockets Layer (SSL) certificate.
  For information about text expressions for SSL and SSL certificate data, see "Default Syntax Expressions: Parsing SSL Certificates" and "Expressions for SSL Certificate Dates."

Note: Parsing a document body, such as the body of a POST request, can affect performance. You may want to test the performance impact of policies that evaluate a document body.

Guidelines for Text Expressions

From a performance standpoint, it typically is best to use protocol-aware functions in an expression. For example, the following expression makes use of a protocol-aware function:

```
HTTP.REQ.URL.QUERY
```

The previous expression performs better than the following equivalent expression, which is based on string parsing:

```
HTTP.REQ.URL.AFTER_STR("?")
```

In the first case, the expression looks specifically at the URL query. In the second case, the expression scans the data for the first occurrence of a question mark.

There is also a performance benefit from structured parsing of text, as in the following expression:

```
HTTP.REQ.HEADER("Example").TYPECAST_LIST_T(',').GET(1)
```

(For more information on typecasting, see "Typecasting Data.") The typecasting expression, which collects comma-delimited data and structures it into a list, typically would perform better than the following unstructured equivalent:

```
HTTP.REQ.HEADER("Example").AFTER_STR(",").BEFORE_STR(",")
```

Finally, unstructured text expressions typically have better performance than regular expressions. For example, the following is an unstructured text expression:

```
HTTP.REQ.HEADER("Example").AFTER_STR("more")
```

The previous expression would generally provide better performance than the following equivalent, which uses a regular expression:

```
HTTP.REQ.HEADER("Example").AFTER_REGEX(re/more/)
```

For more information on regular expressions, see "Regular Expressions."

Expression Prefixes for Text in HTTP Requests and Responses

Oct 07, 2016

An HTTP request or response typically contains text, such as in the form of headers, header values, URLs, and POST body text. You can configure expressions to operate on one or more of these text-based items in an HTTP request or response.

The following table describes the expression prefixes that you can configure to extract text from different parts of an HTTP request or response.

Table 1. HTTP Expression Prefixes That Return Text

Prefix	Description
HTTP.REQ.BODY(<integer>)	Example: HTTP.REQ.BODY(100) It will return first 100 characters of HTTP Request body. If the length of body is less than 100 then whole body will result as output.
HTTP.REQ.HOSTNAME	Example: HTTP.REQ.HOSTNAME.EQ("abc.com") The above example returns true if hostname is abc.com. It returns HTTP Host Name object from this request. If the target hostname is present in the first line of the request then that is selected. Otherwise the value in the last occurrence of the HOST header is selected. The format of output is abc.foo.com:8080. For more information on typecasting, see "Typecasting Data."
HTTP.REQ.HOSTNAME. DOMAIN	Example: HTTP.REQ.URL.HOSTNAME.DOMAIN.EQ("foobar.com") The above example returns true if domain name is foobar.com. It returns Domain name part of the hostname. If the hostname is www.foobar.com or www.foobar.com:8080, then domain is foobar.com.
HTTP.REQ.HOSTNAME. SERVER	Example: HTTP.REQ.URL.HOSTNAME.SERVER.EQ("www.foobar.com") The above example returns true if server name is www.foobar.com. It returns Domain name part of the hostname. If the hostname is www.foobar.com or www.foobar.com:8080, then domain is www.foobar.com.
HTTP.REQ.METHOD	Example: HTTP.REQ.HOSTNAME.SERVER("www.foobar.com") The above example returns true if domain name is www.foobar.com. Correct one is HTTP.REQ.HOSTNAME.SERVER. It returns Domain name part of the hostname.
HTTP.REQ.URL	Example: HTTP.REQ.URL.EQ("http://www.google.com") The above example returns true if domain name is http://www.google.com. It returns the HTTP URL object from request.
HTTP.REQ.URL.HOSTNAME	Example: HTTP.REQ.URL.HOSTNAME.EQ("abc.foo.com:8080") The above example returns true if hostname in URL is abc.foo.com:8080. It returns HTTP Host Name present in the URL. For more information on typecasting, see "Typecasting Data."
HTTP.REQ.URL.HOSTNAME.DOMAIN	Example: HTTP.REQ.URL.HOSTNAME.DOMAIN.EQ("foobar.com") The above example returns true if domain name is foobar.com. It returns Domain

	name part of the hostname. If the hostname is www.foobar.com or www.foobar.com:8080, then domain is foobar.com.
HTTP.REQ.URL.HOSTNAME.SERVER	Example: The above example returns true if server name is www.foobar.com. It returns Domain name part of the hostname. If the hostname is www.foobar.com or www.foobar.com:8080, then domain is www.foobar.com.
HTTP.REQ.URL.HOSTNAME.PORT	Example: HTTP.REQ.URL.HOSTNAME.PORT.EQ(80) The above example returns true if port is 80. It returns number on the port part of the hostname.
HTTP.REQ.URL.PATH	Example:HTTP.REQ.URL.PATH.GET(1) If the URL is http://www.foo.com/a/b/c/bar.html?a=1 then operation will select /a/b/c/bar.html, then the above example will result in “a”. It returns / separated List on the path component of the URL.
HTTP.REQ.URL.PATH_AND_QUERY	Example: HTTP.REQ.URL.PATH_AND_QUERY If the URL is http://www.foo.com/a/b/c/bar.html?a=1 then it will return /a/b/c/bar.html?a=1. It returns the portion of the URL following the hostname
HTTP.REQ.URL.PROTOCOL	Example: HTTP.REQ.URL.PROTOCOL If the URL is http://www.foo.com/a/b/c/bar.html?a=1 then operation will result in HTTP. It results in the protocol present in the URL.
HTTP.REQ.URL.QUERY	Example: HTTP.REQ.URL.QUERY If the URL is http://www.foo.com?abc=1&def=2 will result in abc=1&def=2. It results as Name-Value List (with delimiters = and &) on the query component of the URL.
HTTP.REQ.URL.QUERY.VALUE	Example: -NA- It returns the value component of the specified name-value component in the list.
HTTP.REQ.URL.SUFFIX	Example: HTTP.REQ.URL.SUFFIX If the path is /a/b/c.html then this operation will result html. It returns filename suffice of the URL.
HTTP.REQ.USER	Example: HTTP.REQ.USER.GROUPS('grp1:grp2') The above example will return list on the Group which is separated by given delimiter I.e. “:" It returns the AAA User associated with the current HTTP transaction.
HTTP.REQ.USER.EXTERNAL_GROUPS	Example: HTTP.REQ.USER.EXTERNAL_GROUPS The above example will list external groups which are separated by “,". IT returns list of external groups which is separated by “,".
HTTP.REQ.USER.EXTERNAL_GROUPS.IGNORE_EMPTY_ELEMENTS	Example: HTTP.REQ.USER.EXTERNAL_GROUPS. IGNORE_EMPTY_ELEMENTS.COUNT If AAA User associated with the current HTTP transaction is part of some external groups : 123,,24, ,15 then HTTP.REQ.USER.EXTERNAL_GROUPS. IGNORE_EMPTY_ELEMENTS.COUNT gives 4,

	whereas HTTP.REQ.USER.EXTERNAL_GROUPS.COUNT gives 5. It ignores empty elements in the list.
HTTP.REQ.USER.EXTERNAL_GROUPS(sep)	Example: HTTP.REQ.USER.EXTERNAL_GROUPS(":") The above example will list external groups which are separated by ":" It returns list of external groups which is separated by given delimiter.
HTTP.REQ.USER.GROUPS	Example: HTTP.REQ.USER.GROUPS The above example will list groups which are separated by ",". IT returns list of groups which is separated by ",".
HTTP.REQ.USER.GROUPS.IGNORE_EMPTY_ELEMENTS	Example: HTTP.REQ.USER.GROUPS. IGNORE_EMPTY_ELEMENTS.COUNT If AAA User associated with the current HTTP transaction is part of some groups : 123,,24, ,15 then HTTP.REQ.USER.GROUPS. IGNORE_EMPTY_ELEMENTS.COUNT gives 4, whereas HTTP.REQ.USER.GROUPS.COUNT gives 5. It ignores empty elements in the list.
HTTP.REQ.USER.GROUPS(sep)	Example: HTTP.REQ.USER.GROUPS(":") The above example will list groups which are separated by ":" IT returns list of groups which is separated by given delimiter.
HTTP.REQ.USER.INTERNAL_GROUPS	Example: HTTP.REQ.USER.INTERNAL_GROUPS The above example will list internal groups which are separated by ",". IT returns list of internal groups which is separated by ",".
HTTP.REQ.USER.INTERNAL_GROUPS.IGNORE_EMPTY_ELEMENTS	Example: HTTP.REQ.USER.INTERNAL_GROUPS. IGNORE_EMPTY_ELEMENTS.COUNT If AAA User associated with the current HTTP transaction is part of some internal groups : 123,,24, ,15 then HTTP.REQ.USER.INTERNAL_GROUPS.IGNORE_EMPTY_ELEMENTS.COUNT gives 4, whereas HTTP.REQ.USER.INTERNAL_GROUPS.COUNT gives 5. It ignores empty elements in the list.
HTTP.REQ.USER.INTERNAL_GROUPS(sep)	Example: HTTP.REQ.USER.INTERNAL_GROUPS(":") The above example will list internal groups which are separated by ":" IT returns list of internal groups which is separated by given delimiter.
HTTP.REQ.USER.IS_MEMBER_OF(group_name)	Example: HTTP.REQ.USER.IS_MEMBER_OF(grp1) The above example returns true is the current AAA user is member of group grp1. It returns TRUE if the user is a member of the group group_name.
HTTP.REQ.USER.NAME	Example: HTTP.REQ.USER.NAME The above example will return name of the user. It returns the name of user. This is the name used by user for login unless it is overridden by name from external authentication server.
HTTP.REQ.USER.PASSWD	Example: HTTP.REQ.USER.PASSWD The above example will return password of the user. It returns the password of user.

HTTP.REQ.VERSION	Example: HTTP.REQ.VERSION The above example returns HTTP version information.
HTTP.RES.BODY(<integer>)	Example: HTTP.RES.BODY(100) It will return first 100 characters of HTTP Response body. If the length of body is less than 100 then whole body will result as output.
HTTP.RES.STATUS_MSG	Example: HTTP.RES.STATUS_MSG The above example results status message of response. It can be “Done”, some error etc.
HTTP.RES.VERSION	Example: HTTP.RES.VERSION The above example returns HTTP version information.
HTTP.REQ.URL.HOSTNAME.EQ(<hostname>)	Example: HTTP.REQ.URL.HOSTNAME.EQ("abc.foo.com:8080") The above example returns true if hostname in URL is abc.foo.com:8080. It returns HTTP Host Name present in the URL.
HTTP.REQ.IS_NTLM_OR_NEGOTIATE	Example: HTTP.REQ.IS_NTLM_OR_NEGOTIATE The above example returns TRUE if request is part of NTLM or NEGOTIATE connection.
HTTP.REQ.URL.PATH.IGNORE_EMPTY_ELEMENTS	Example: HTTP.REQ.URL.PATH. IGNORE_EMPTY_ELEMENTS.COUNT If request URL has path (/123//24//15) elements as : 123,,24, ,15 then HTTP.REQ.URL.PATH.IGNORE_EMPTY_ELEMENTS.COUNT gives 4, whereas HTTP.REQ.URL.PATHS.COUNT gives 5. It ignores empty elements in the list.
HTTP.REQ.URL.QUERY.IGNORE_EMPTY_ELEMENTS	Example: HTTP.REQ.URL.QUERY.IGNORE_EMPTY_ELEMENTS.COUNT If request URL has path as : abc=1&&def=2&g=3&h=6 then HTTP.REQ.URL.QUERY.IGNORE_EMPTY_ELEMENTS.COUNT gives 4, whereas HTTP.REQ.URL.QUERY.COUNT gives 5. It ignores empty elements in the list.

Expression Prefixes for VPNs and Clientless VPNs

Aug 30, 2013

The default syntax expression engine provides prefixes that are specific to parsing VPN or Clientless VPN data. This data includes the following:

- Host names, domains, and URLs in VPN traffic.
- Protocols in the VPN traffic.
- Queries in the VPN traffic.

These text elements are often URLs and components of URLs. In addition to applying the text-based operations on these elements, you can parse these elements by using operations that are specific to parsing URLs. For more information, see "Expressions for Extracting Segments of URLs."

The following table describes the expression prefixes for this type of data.

Table 1. VPN and Clientless VPN Expression Prefixes That Return Text

VPN and Clientless VPN Expression	Description
VPN.BASEURL.CVPN_DECODE	Extracts the original URL from a clientless VPN URL.
VPN.BASEURL.CVPN_ENCODE	Converts a URL to clientless VPN format.
VPN.BASEURL.HOSTNAME	Extracts the HTTP host name from the host name in the URL. This prefix cannot be used in bidirectional policies.
VPN.BASEURL.HOSTNAME.DOMAIN	Extracts the domain name from the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, this prefix extracts mycompany.com. This prefix returns incorrect results if the host name is an IP address. For information on expressions for IP addresses, see "Default Syntax Expressions: IP and MAC Addresses, Throughput, VLAN IDs." All text operations after this prefix are case insensitive.
VPN.BASEURL.HOSTNAME.EQ (<hostname>)	Returns a Boolean TRUE if the host name matches <hostname>. The comparison is case insensitive. For example, if the host name is www.mycompany.com, the following returns TRUE: vpn.baseurl.hostname.eq("www.mycompany.com") If the text mode is URLENCODED, the host name is decoded before comparison. For more information, see "Operations for HTTP, HTML, and XML Encoding and "Safe" Characters."
VPN.BASEURL.HOSTNAME.SERVER	Evaluates the server portion of the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the server is www.mycompany.com. All text operations after this prefix are case insensitive.
VPN.BASEURL.PATH	Extracts a slash- (/) separated list from the path component of the URL. For example, this prefix extracts /a/b/c/mypage.html from the following URL: http://www.mycompany.com/a/b/c/mypage.html?a=1 The following expression selects just the "a": http.req.url.path.get(1) For more information on the GET operation, see "Expressions for Extracting Segments of URLs."
VPN.BASEURL.PATH.IGNORE_EMPTY_ELEMENTS	This prefix ignores the elements in a list. For example, the following comma-separated list has an empty element after "a=10": a=10,,b=11, ,c=89 The element following b=11 contains a space, and by default, is not considered an empty element.

VPN and Clientless VPN Expression	Description
	Consider the following HTTP header: Cust_Header : 123,,24, ,15 The following expression returns a count of 4 when evaluating this header: http.req.header("Cust_Header").typecase_list_t(',').ignore_empty_elements.count The following expression returns a count of 5 when evaluating this header: http.req.header("Cust_Header").typecase_list_t(','). count
VPN.BASEURL.PATH_AND_QUERY	Evaluates the text in the URL that follows the host name. For example, if the URL is http://www.mycompany.com/a/b/c/mypage.html?a=1, this prefix evaluates /a/b/c/mypage.html?a=1.
VPN.BASEURL.PROTOCOL	Evaluates the protocol in the URL. Do not use this prefix in bidirectional policies.
VPN.BASEURL.QUERY	Extracts a name-value list, using the "=" and "&" delimiters from the query string in a URL.
VPN.BASEURL.QUERY.IGNORE_EMPTY_ELEMENTS	This method ignores the empty elements in a name-value list. For example, in the following name-value list, there is an empty element following "a=10": a=10;;b=11; ;c=89 The element following b=11 contains a space and is not considered an empty element. Consider the following HTTP header: Cust_Header : a=1;;b=2; ;c=3 The following expression produces a count of 4 after evaluating this header: http.req.header("Cust_Header").typecast_nvlist_t('=',';').ignore_empty_elements.count The following expression produces a count of 5 after evaluating the header: http.req.header("Cust_Header").typecast_nvlist_t('=',';').
VPN.BASEURL.SUFFIX	Evaluates the file name suffix in a URL. For example, if the path is /a/b/c/my.page.html, this operation selects "html."
VPN.CLIENTLESS_BASEURL	Evaluates the clientless VPN base URL.
VPN.CLIENTLESS_BASEURL.CVPN_DECODE	Extracts the original URL from the clientless VPN formatted URL.
VPN.CLIENTLESS_BASEURL.CVPN_ENCODE	Converts a URL to the clientless VPN format.
VPN.CLIENTLESS_BASEURL.HOSTNAME	Evaluates the host name in the URL. Do not use this prefix in bidirectional policies.
VPN.CLIENTLESS_BASEURL.HOSTNAME.DOMAIN	Evaluates the domain name part of the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the domain is mycompany.com. This operation returns incorrect results if the host name is an IP address. For information on expressions for IP addresses, see "Default Syntax Expressions: IP and MAC Addresses, Throughput, VLAN IDs." All text operations after this prefix are case insensitive.
VPN.CLIENTLESS_BASEURL.HOSTNAME.EQ(<hostname>)	Returns a Boolean TRUE if the host name matches <hostname>. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the following

VPN and Clientless VPN Expression	Description
	is true: vpn.clientless_baseurl. hostname.eq("www.mycompany.com") The comparison is case insensitive. If the textmode is URLENCODED, the host name is decoded before comparison. For more information, see "Operations for HTTP, HTML, and XML Encoding and "Safe" Characters."
VPN.CLIENTLESS_BASEURL.HOSTNAME.SERVER	Evaluates the server part of a host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the server is www.mycompany.com. All text operations after this prefix are case insensitive.
VPN.CLIENTLESS_BASEURL.PATH	Evaluates a slash- (/) separated list in the URL path. For example, this prefix selects /a/b/c/mypage.html from the following URL: http://www.mycompany.com/a/b/c/mypage.html?a=1 The following expression selects "a" from the preceding URL: http.req.url.path.get(1) For more information on the GET operation, see "Expressions for Extracting Segments of URLs."
VPN.CLIENTLESS_BASEURL.PATH.IGNORE_EMPTY_ELEMENTS	Ignores empty elements in a list. For example, if the list delimiter is a comma (,) the following list has an empty element following "a=10": a=10,b=11, ,c=89 The element following b=11 contains a space and is not considered an empty element. Consider the following HTTP header: Cust_Header : 123,,24, ,15 The following expression returns a value of 4 after evaluating this header: http.req.header("Cust_Header").typecast_list_t(',').ignore_empty_elements.count The following expression returns a value of 5 after evaluating this header: http.req.header("Cust_Header").typecast_list_t(',').
VPN.CLIENTLESS_BASEURL.PATH_AND_QUERY	Evaluates the text following the host name in a URL. For example, this prefix selects /a/b/c/mypage.html?a=1 from the following URL: http://www.mycompany.com/a/b/c/mypage.html?a=1
VPN.CLIENTLESS_BASEURL.PROTOCOL	Evaluates the protocol in the URL. Do not use this prefix in bidirectional policies.
VPN.CLIENTLESS_BASEURL.QUERY	Extracts a name-value list that uses the delimiters "=" and "&" from a URL query string.
VPN.CLIENTLESS_BASEURL.QUERY.IGNORE_EMPTY_ELEMENTS	Ignores empty elements in a name-value list. For example, the following list contains an empty element after "a=10": a=10;;b=11; ;c=89 The element following b=11 contains a space and is not considered an empty element. As another example, consider the following http header: Cust_Header : a=1;;b=2; ;c=3 The following expression returns a value of 4 after evaluating the preceding header: http.req.header("Cust_Header").typecast_nvlist_t('=',';').ignore_empty_elements.count The following expression returns a value of 5 after evaluating the preceding header:

VPN and Clientless VPN Expression	Description
	http.req.header("Cust_Header").typecast_nvlist_t('=',';')
VPN.CLIENTLESS_BASEURL.SUFFIX	Evaluates the file suffix in a URL. For example, if the URL path is /a/b/c/mypage.html then this operation selects html.
VPN.CLIENTLESS_HOSTURL	Selects the clientless VPN host URL.
VPN.CLIENTLESS_HOSTURL.CVPN_DECODE	Selects the original URL from the clientless VPN formatted URL.
VPN.CLIENTLESS_HOSTURL.CVPN_ENCODE	Converts a URL to clientless VPN format.
VPN.CLIENTLESS_HOSTURL.HOSTNAME	Extracts the host name in the URL. Do not use this prefix in bidirectional policies.
VPN.CLIENTLESS_HOSTURL.HOSTNAME.DOMAIN	Extracts the domain name from the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the domain is mycompany.com. This operation returns incorrect results if the host name contains an IP address. For information on expressions for IP addresses, see "Default Syntax Expressions: IP and MAC Addresses, Throughput, VLAN IDs." All text operations after this prefix are case insensitive.
VPN.CLIENTLESS_HOSTURL.HOSTNAME.EQ(<hostname>)	Results in Boolean TRUE if the host name matches the <hostname> argument. The comparison is case insensitive. For example, if the host name is www.mycompany.com or www.mycompany.com., the following expression returns TRUE: vpn.clilentless_hosturl. hostname.eq("www.mycompany.com") If the text mode is URLENCODED, the host name is decoded before comparison. For more information, see "Operations for HTTP, HTML, and XML Encoding and “Safe” Characters."
VPN.CLIENTLESS_HOSTURL.HOSTNAME.SERVER	Evaluates the server part of the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the server is www.mycompany.com. The comparison is case insensitive, and all text operations after this method are case insensitive.
VPN.CLIENTLESS_HOSTURL.PATH	Evaluates a slash- (/) separated list on the path component of the URL. For example, consider the following URL: http://www.mycompany.com/a/b/c/mypage.html?a=1 This prefix selects /a/b/c/mypage.html from the preceding URL.
VPN.CLIENTLESS_HOSTURL.PATH.IGNORE_EMPTY_ELEMENTS	This method ignores the empty elements in a list. For example, if the delimiter in a list is “,” the following list contains an empty element after the entry “a=10”: a=10,b=11, ,c=89 The element following b=11 contains a space and is not considered an empty element. Consider the following header: Cust_Header : 123,,24, ,15 The following expression returns a value of 4 for this header: http.req.header("Cust_Header").typecast_list_t(','). ignore_empty_elements.count The following expression returns a value of 5 for the same header: http.req.header("Cust_Header").typecast_list_t(',').

VPN and Clientless VPN Expression	Description
VPN.CLIENTLESS_HOSTURL.PATH_AND_QUERY	Evaluates the portion of the URL that follows the host name. For example, consider the following URL: http://www.mycompany.com/a/b/c/mypage.html?a=1 This prefix returns /a/b/c/mypage.html?a=1 from the preceding URL.
VPN.CLIENTLESS_HOSTURL.PROTOCOL	Evaluates the protocol in the URL. Do not use this prefix in bidirectional policies.
VPN.CLIENTLESS_HOSTURL.QUERY	Extracts a name-value list, using the "=" and "&" delimiters from a URL query string.
VPN.CLIENTLESS_HOSTURL.QUERY.IGNORE_EMPTY_ELEMENTS	Ignores empty elements in a name-value list. For example, the following list uses a semicolon (;) delimiter. This list contains an empty element after "a=10": a=10;;b=11; ;c=89 In the preceding example, the element following b=11 is not considered an empty element. Consider the following header: Cust_Header : a=1;;b=2; ;c=3 The following expression returns a value of 4 after evaluating this header: http.req.header("Cust_Header").typecast_nvlist_t('=',';').ignore_empty_elements.count The following expression returns a value of 5 after evaluating the same header: http.req.header("Cust_Header").typecast_nvlist_t('=',';')
VPN.CLIENTLESS_HOSTURL.SUFFIX	Extracts a file name suffix in a URL. For example, if the path is /a/b/c/my.page.html, this prefix selects html.
VPN.HOST.DOMAIN	Extracts the domain name part of the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the domain is mycompany.com. This prefix returns incorrect results if the host name contains an IP address. For information on expressions for IP addresses, see "Default Syntax Expressions: IP and MAC Addresses, Throughput, VLAN IDs." All text operations after this prefix case insensitive.
VPN.HOST.EQ(<hostname>)	Returns a Boolean TRUE value if the host name matches the <hostname>. The comparison is case insensitive. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the following returns TRUE: vpn.host.eq("www.mycompany.com") If the text mode is URLENCODED the host name is decoded before comparison. For more information, see "Operations for HTTP, HTML, and XML Encoding and "Safe" Characters."
VPN.HOST.SERVER	Extracts the server name part of the host name. For example, if the host name is www.mycompany.com or www.mycompany.com:8080, the server is www.mycompany.com. All text operations after this prefix are case insensitive.

Basic Operations on Text

Aug 30, 2013

Basic operations on text include operations for string matching, calculating the length of a string, and controlling case sensitivity. You can include white space in a string that is passed as an argument to an expression, but the string cannot exceed 255 characters.

The following table lists basic string matching operations in which the functions return a Boolean TRUE or FALSE.

Table 1. String Comparison Functions

Function	Description
<text>.CONTAINS(<string>)	Returns a Boolean TRUE value if the target contains <string>. Following is an example: `http.req.url.contains(".jpeg")`
<text>.EQ(<string>)	Returns a Boolean TRUE value if the target is an exact match with <string>. For example, the following expression returns a Boolean TRUE for a URL with a host name of “myhostabc”: `http.req.url.hostname.eq("myhostabc")`
<text>.STARTSWITH(<string>)	Returns a Boolean TRUE value if the target begins with <string>. For example, the following expression returns a Boolean TRUE for a URL with a host name of “myhostabc”: `http.req.url.hostname.startswith("myhost")`
<text>.ENDSWITH(<string>)	Returns a Boolean TRUE value if the target ends with <string>. For example, the following expression returns a Boolean TRUE for a URL with a host name of “myhostabc”: `http.req.url.hostname.endswith("abc")`
<text>.NE(<string>)	Returns a Boolean TRUE value if the prefix is not equal to the string argument. If the prefix returns a non-string value, the function argument is compared to the string representation of the value returned by the prefix. You can use the functions with SET_TEXT_MODE(IGNORECASE) or

Function	Description
	SET_TEXT_MODE(NOIGNORECASE), and with both ASCII and UTF-8 character sets.
<text>.GT(<string>)	Returns a Boolean TRUE value if the prefix is alphabetically greater than the string argument. If the prefix returns a non-string value, the function argument is compared to the string representation of the value returned by the prefix. You can use the functions with SET_TEXT_MODE(IGNORECASE) or SET_TEXT_MODE(NOIGNORECASE), and with both ASCII and UTF-8 character sets.
<text>.GE(<string>)	Returns a Boolean TRUE value if the prefix is alphabetically greater than or equal to the string argument. If the prefix returns a non-string value, the function argument is compared to the string representation of the value returned by the prefix. You can use the functions with SET_TEXT_MODE(IGNORECASE) or SET_TEXT_MODE(NOIGNORECASE), and with both ASCII and UTF-8 character sets.
<text>.LT(<string>)	Returns a Boolean TRUE value if the prefix is alphabetically lesser than the string argument. If the prefix returns a non-string value, the function argument is compared to the string representation of the value returned by the prefix. You can use the functions with SET_TEXT_MODE(IGNORECASE) or SET_TEXT_MODE(NOIGNORECASE), and with both ASCII and UTF-8 character sets.
<text>.LE(<string>)	Returns a Boolean TRUE value if the prefix is alphabetically lesser than or equal to the string argument. If the prefix returns a non-string value, the function argument is compared to the string representation of the value returned by the prefix. You can use the functions with SET_TEXT_MODE(IGNORECASE) or SET_TEXT_MODE(NOIGNORECASE), and with both ASCII and UTF-8 character sets.

The <text>.LENGTH operation returns a numeric value that is equal to the number of characters (not bytes) in a string:

<text>.LENGTH

For example, you may want to identify request URLs that exceed a particular length. Following is an expression that

implements this example:

HTTP.REQ.URL.LENGTH < 500

After taking a count of the characters or elements in a string, you can apply numeric operations to them. For more information, see "Default Syntax Expressions: Working with Dates, Times, and Numbers."

The following functions operate on the case (upper-case or lower-case) of the characters in the string.

Table 2. Functions for Considering, Ignoring, and Changing Text Case

Function	Description
<text>.SET_TEXT_MODE(IGNORECASE|NOIGNORECASE)	This function turns case sensitivity on or off for all text operations.
<text>.TO_LOWER	Converts the target to lowercase for a text block of up to 2 kilobyte (KB). Returns UNDEF if the target exceeds 2 KB. For example, the string “ABCd:” is converted to “abcd:”.
<text>.TO_UPPER	Converts the target to uppercase. Returns UNDEF if the target exceeds 2 KB. For example, the string “abcD:” is converted to “ABCD:”.

You can use the STRIP_CHARS(<string>) function to remove specific characters from the text that is returned by a default syntax expression prefix (the input string). All instances of the characters that you specify in the argument are stripped from the input string. You can use any text method on the resulting string, including the methods used for matching the string with a pattern set.

For example, in the expression CLIENT.UDP.DNS.DOMAIN.STRIP_CHARS(".-_"), the STRIP_CHARS(<string>) function strips all periods (.), hyphens (-), and underscores (_) from the domain name returned by the prefix CLIENT.UDP.DNS.DOMAIN. If the domain name that is returned is "a.dom_ai_n-name", the function returns the string "adomainname".

In the following example, the resulting string is compared with a pattern set called "listofdomains":

CLIENT.UDP.DNS.DOMAIN.STRIP_CHARS(".-_").CONTAINS_ANY("listofdomains")

Note: You cannot perform a rewrite on the string that is returned by the STRIP_CHARS(<string>) function.
The following functions strip matching characters from the beginning and end of a given string input.

Table 3. Functions for Stripping Characters From the Beginning or End of a String

Function	Description
<text>.STRIP_START_CHARS(s)	Strips matching characters from the beginning of the input string until the first non-matching character is found and returns the remainder of the string. You must specify the characters that you want to strip as a single string within quotation marks. For example, if the name of a header is TestLang and ://_en_us: is its value, HTTP.RES.HEADER("TestLang").STRIP_START_CHARS(":/_") strips the specified characters from the beginning of the value of the header until the first non-matching character e is found and returns en_us: as a string.
<text>.STRIP_END_CHARS(s)	Strips matching characters from the end of the input string to the first non-matching character is found and returns the remainder of the string. You must specify the characters that you want to strip as a single string within quotation marks. For example, if the name of a header is TestLang and ://_en_us: is its value, HTTP.RES.HEADER("TestLang").STRIP_START_CHARS(":/_") strips the specified characters from the end of the value of the header until the first non-matching character s is found and returns ://_en_us as a string.

You can use the APPEND() function to append the string representation of the argument to the string representation of the value returned by the preceding function. The preceding function can be one that returns a number, unsigned long, double, time value, IPv4 address, or IPv6 address. The argument can be a text string, number, unsigned long, double, time value, IPv4 address, or IPv6 address. The resulting string value is the same string value that is obtained by using the + operator.

Complex Operations on Text

May 25, 2015

In addition to performing simple string matching, you can configure expressions that examine more complex aspects of text, including examining the length of a string and looking within a text block for patterns rather than specific strings.

Be aware of the following for any text-based operation:

- For any operation that takes a string argument, the string cannot exceed 255 characters.
- You can include white space when you specify a string in an expression.

This document includes the following details:

The following operations extract strings on the basis of a character count.

Table 1. String Operations Based on a Character Count

Character Count Operation	Description
<text>.TRUNCATE(<count>)	Returns a string after truncating the end of the target by the number of characters in <count>. If the entire string is shorter than <count>, nothing is returned.
<text>.TRUNCATE(<character>, <count>)	Returns a string after truncating the text after <character> by the number of characters specified in <count>.
<text>.PREFIX(<character>, <count>)	Selects the longest prefix in the target that has at most <count> occurrences of <character>.
	<text>.SUFFIX(<character>, <count>) Selects the longest suffix in the target that has at most <count> occurrences of <character>. For example, consider the following response body: JLEwx The following expression returns a value of “JLEwx”: http.res.body(100).suffix('L',1) The following expression returns “LLEwx”: http.res.body(100).suffix('L',2)
<text>.SUBSTR(<starting_offset>,	Select a string with <length> number of characters from the target object. Begin extracting the

Character Count Operation	Description
<text>.SKIP(<character>, <count>)	Select a string from the target after skipping over the longest prefix that has at most <count> occurrences of <character>.

You can extract a subset of a larger string by using one of the operations in the following table.

Table 2. Basic Operations on a Portion of a String

Basic Text Operation	Description
<text>.BEFORE_STR(<string>)	Returns the text that precedes the first occurrence of <string>. If there is no match for <string>, the expression returns a text object of 0 length. Following is an example: http.res.body(1024).after_str("start_string").before_str("end_string").contains("https")
<text>.AFTER_STR(<string>)	Returns the text that follows the first occurrence of <string>. If there is no match for <string>, the expression returns a text object of 0 length. Following is an example: http.res.body(1024).after_str("start_string").before_str("end_string").contains("https")
<text>.BETWEEN(<starting string>, <ending string>)	Returns a Boolean TRUE value if the length of the text object is greater than or equal to the sum <starting string>, <ending string> argument lengths, and if a prefix of the target matches <starting string>, and if the suffix of the target matches <ending string>.
<text>.PREFIX(<prefix length>)	Returns the starting string from a target block of text that contains the number of characters in the <prefix length> argument. If the <prefix length> argument exceeds the number of characters in the target, the entire string is selected.
<text>.SUFFIX(<suffix length>)	Returns the ending string from a target block of text that contains the number of characters in the <suffix length> argument. If the <suffix length> argument exceeds the number characters in the target, the entire string is selected.
<text>.SUBSTR(<string>)	Select the first block of text in the target that matches the <string>.
<text>.SKIP(<prefix length>)	Selects the text in the target after skipping over a <prefix length> number of characters. If the entire target has fewer characters than <prefix length>, the entire target is skipped.
<text>.STRIP_END_WS	Selects the text after removing white space from the end of the target.
<text>.STRIP_START_WS	Selects the text after removing white space from the beginning of the target.

Basic Text Operation	Description
<text>.UNQUOTE(<character>)	Selects the <character>, removes white space that immediately precedes and follows the <character>, and if the remaining text is quoted by <character>, this prefix also removes the quotes. For example, the operation UNQUOTE("") changes the following text: " abc xyz def " To the following: abc xyz def

The COMPARE operation examines the first nonmatching character of two different strings. This operation is based on lexicographic order, which is the method used when ordering terms in dictionaries.

This operation returns the arithmetic difference between the ASCII values of the first nonmatching characters in the compared strings. The following differences are examples:

- The difference between “abc” and “abd” is -1 (based on the third pair-wise character comparison).
- The difference between “@” and “abc” is -33.
- The difference between “1” and “abc” is -47.

Following is the syntax for the COMPARE operation.

<text>.COMPARE(<string>)

You can use the following functions to treat a string of bytes that represent text as a sequence of bytes, extract 8, 16, or 32 bits from the sequence, and then convert the extracted bits to an integer.

Table 3. Operations for Extracting an Integer from a String of Bytes That Represent Text

Function	Description
<text>.GET_SIGNED8(<n>)	Treats the string of bytes represented by text as a sequence of 8-bit signed integers and returns the integer at byte offset n. If the offset makes all or part of the value outside of the current text, an UNDEF condition is raised.
<text>.GET_UNSIGNED8(<n>)	Treats the string of bytes represented by text as a sequence of 8-bit unsigned integers and returns the integer at byte offset n. If the offset makes all or part of the value outside of the current text, an UNDEF condition is raised.
<text>.GET_SIGNED16(<n>, <endianness>)	Treats the text string returned by the prefix as a string of bytes, extracts 16 bits starting at byte offset n, and converts the extracted bit sequence to a 16-bit signed integer. If the offset makes all or part of the value outside of the current text, an UNDEF condition is raised. The first parameter n is the byte offset from the current position in the text string. Providing a byte offset enables the function to handle items that are not aligned on the boundaries that are required by indexes. The second parameter, endianness, takes a mnemonic value of LITTLE_ENDIAN or BIG_ENDIAN. Note: In NetScaler 9.2, the parameter n was an index into an array of 16-bit items. In NetScaler 9.3, the parameter is a byte offset. Therefore, if you used this function in NetScaler 9.2, after you upgrade to NetScaler 9.3, you must change n to 2*n to obtain the same results as you did earlier. For example, if the value of n before the upgrade was 4, you must change the value of n to 8. The

Function	Description
	parameter endianness also no longer takes the values that it did in NetScaler 9.2, which were 0 and 1. Instead, endianness accepts the mnemonic values mentioned earlier. **Example** HTTP.REQ.BODY(100).GET_SIGNED16(8, BIG_ENDIAN)
<text>.GET_UNSIGNED16(<n>, <endianness>)	Treats the text string returned by the prefix as a string of bytes, extracts 16 bits starting at byte offset n, and converts the extracted bit sequence to a 16-bit unsigned integer. If the offset makes all or part of the value outside of the current text, an UNDEF condition is raised. The first parameter n is the byte offset from the current position in the text string. Providing a byte offset enables the function to handle items that are not aligned on the boundaries that are required by indexes. The second parameter, endianness, takes a mnemonic value of LITTLE_ENDIAN or BIG_ENDIAN. Note: In NetScaler 9.2, the parameter n was an index into an array of 16-bit items. In NetScaler 9.3, the parameter is a byte offset. Therefore, if you used this function in NetScaler 9.2, after you upgrade to NetScaler 9.3, you must change n to 2*n to obtain the same results as you did earlier. For example, if the value of n before the upgrade was 4, you must change the value of n to 8. The parameter endianness also no longer takes the values that it did in NetScaler 9.2, which were 0 and 1. Instead, endianness accepts the mnemonic values mentioned earlier. **Example** HTTP.REQ.BODY(100).GET_UNSIGNED16(8, LITTLE_ENDIAN)
<text>.GET_SIGNED32(<n>, <endianness>)	Treats the text string returned by the prefix as a string of bytes, extracts 32 bits starting at byte offset n, and converts the extracted bit sequence to a 32-bit signed integer. If the offset makes all or part of the value outside of the current text, an UNDEF condition is raised. The first parameter n is the byte offset from the current position in the text string. Providing a byte offset enables the function to handle items that are not aligned on the boundaries that are required by indexes. The second parameter, endianness, takes a mnemonic value of LITTLE_ENDIAN or BIG_ENDIAN. Note: In NetScaler 9.2, the parameter n was an index into an array of 32-bit items. In NetScaler 9.3, the parameter is a byte offset. Therefore, if you used this function in NetScaler 9.2, after you upgrade to NetScaler 9.3, you must change n to 4*n to obtain the same results as you did earlier. For example, if the value of n before the upgrade was 4, you must change the value of n to 16. The parameter endianness also no longer takes the values that it did in NetScaler 9.2, which were 0 and 1. Instead, endianness accepts the mnemonic values mentioned earlier. **Example** HTTP.REQ.BODY(1000).GET_SIGNED32(12, BIG_ENDIAN)
<text>.GET_UNSIGNED32(<n>, <endianness>)	Treats the text string returned by the prefix as a string of bytes, extracts 32 bits starting at byte offset n, and returns the extracted bit sequence as part of a 64-bit unsigned long integer. If the offset makes all or part of the value outside of the current text, an UNDEF condition is raised. The first parameter n is the byte offset from the current position in the text string. Providing a byte offset enables the function to handle items that are not aligned on the boundaries that are required by indexes. The second parameter, endianness, takes a mnemonic value of LITTLE_ENDIAN or BIG_ENDIAN. **Example** HTTP.REQ.BODY(1000).GET_UNSIGNED32(30, LITTLE_ENDIAN)

Function	Description

You can convert a text string to a hash value by using the HASH function. This function returns a 31-bit positive integer as a result of the operation. Following is the format of the expression:

<text>.HASH

This function ignores case and white spaces. For example, after the operation, the two strings Ab c and a bc would produce the same hash value.

The following two functions encode and decode a text string by applying the Base64 encoding algorithm

Table 4. Functions for Encoding and Decoding a Text String by Using Base64 Encoding

Function	Description
text.B64ENCODE	Encodes the text string (designated by text) by applying the Base64 encoding algorithm.
text.B64DECODE	Decodes the Base64-encoded string (designated by text) by applying the Base64 decoding algorithm. The operation raises an UNDEF if text is not in B64-encoded format.

The EXTEND function is used in rewrite actions that specify patterns or pattern sets and target the bodies of HTTP packets. When a pattern match is found, the EXTEND function extends the scope of the search by a predefined number of bytes on both sides of the matching string. A regular expression can then be used to perform a rewrite on matches in this extended region. Rewrite actions that are configured with the EXTEND function perform rewrites faster than rewrite actions that evaluate entire HTTP bodies using only regular expressions.

The format of the EXTEND function is EXTEND(m,n), where m and n are the number of bytes by which the scope of the search is extended before and after the matching pattern, respectively. When a match is found, the new search scope comprises m bytes that immediately precede the matching string, the string itself, and the n bytes that follow the string. A regular expression can then be used to perform a rewrite on a portion of this new string.

The EXTEND function can be used only if the rewrite action in which it is used fulfills the following requirements:

- The search is performed by using patterns or patterns sets (not regular expressions)
- The rewrite action evaluates only the bodies of HTTP packets.

Additionally, the EXTEND function can be used only with the following types of rewrite actions:

- replace_all
- insert_after_all
- delete_all
- insert_before_all

For example, you might want to delete all instances of "http://exampleurl.com/" and "http://exampleurl.au/" in the first 1000 bytes of the body. To do this, you can configure a rewrite action to search for all instances of the string exampleurl, extend the scope of the search on both sides of the string when a match is found, and then use a regular expression to perform the rewrite in the extended region. The following example extends the scope of the search by 20 bytes to the left and 50 bytes to the right of the matching string:

```
add rewrite action delurl_example delete_all 'HTTP.REQ.BODY(1000)' -pattern exampleurl -refineSearch
'extend(20,50).regex_select(re#http://exampleurl.(com|au)#)'
```

The following function converts text to hexadecimal format and extracts the resulting string:

<text>.BLOB_TO_HEX(<string>)

For example, this function converts the byte string “abc” to “61:62:63”.

Updated: 2013-09-02

In default syntax expressions, you can use the ENCRYPT and DECRYPT functions to encrypt and decrypt text. Data encrypted by the ENCRYPT function on a given NetScaler appliance or high availability (HA) pair is intended for decryption by the DECRYPT function on the same NetScaler appliance or HA pair. The appliance supports the RC4, DES3, AES128, AES192, and AES256 encryption methods. The key value that is required for encryption is not user-specifiable. When an encryption method is set, the appliance automatically generates a random key value that is appropriate for the specified method. The default method is AES256 encryption, which is the most secure encryption method and the one that Citrix recommends.

You do not need to configure encryption unless you want to change the encryption method or you want the appliance to generate a new key value for the current encryption method.

Note: You can also encrypt and decrypt XML payloads. For information about the functions for encrypting and decrypting XML payloads, see "Encrypting and Decrypting XML Payloads."

Configuring Encryption

Updated: 2013-09-02

During startup, the appliance runs the set ns encryptionParams command with, by default, the AES256 encryption method, and uses a randomly generated key value that is appropriate for AES256 encryption. The appliance also encrypts the key value and saves the command, with the encrypted key value, to the NetScaler configuration file. Consequently, the AES256 encryption method is enabled for the ENCRYPT and DECRYPT functions by default. The key value that is saved in the configuration file persists across reboots even though the appliance runs the command each time you restart it.

You can run the set ns encryptionParams command manually, or use the configuration utility, if you want to change the encryption method or if you want the appliance to generate a new key value for the current encryption method. To use the CLI to change the encryption method, set only the method parameter, as shown in "**Example 1: Changing the Encryption Method**." If you want the appliance to generate a new key value for the current encryption method, set the method parameter to the current encryption method and the keyValue parameter to an empty string (""), as shown in "**Example 2: Generating a New Key Value for the Current Encryption Method**." After you generate a new key value, you must save the configuration. If you do not save the configuration, the appliance uses the newly generated key value only until the next restart, after which it reverts to the key value in the saved configuration.

1. Navigate to System > Settings.
2. In the Settings area, click Change Encryption parameters.
3. In the Change Encryption Parameters dialog box, do one of the following:
   - To change the encryption method, in the Method list, select the encryption method that you want.
   - To generate a new key value for the current encryption method, click Generate a new key for the selected method.
4. Click OK.

Using the ENCRYPT and DECRYPT Functions

You can use the ENCRYPT and DECRYPT functions with any expression prefix that returns text. For example, you can use the ENCRYPT and DECRYPT functions in rewrite policies for cookie encryption. In the following example, the rewrite actions encrypt a cookie named MyCookie, which is set by a back-end service, and decrypt the same cookie when it is returned by a client:

```
add rewrite action my-cookie-encrypt-action replace "HTTP.RES.SET_COOKIE.COOKIE(\"MyCookie\").VALUE(0)"
"HTTP.RES.SET_COOKIE.COOKIE(\"MyCookie\").VALUE(0).ENCRYPT" -bypassSafetyCheck YES

add rewrite action my-cookie-decrypt-action replace "HTTP.REQ.COOKIE.VALUE(\"MyCookie\")"
"HTTP.REQ.COOKIE.VALUE(\"MyCookie\").DECRYPT" -bypassSafetyCheck YES
```

After you configure policies for encryption and decryption, save the configuration to bring the policies into effect.

Default Syntax Expressions: Working with Dates, Times, and Numbers

Sep 02, 2013

Most numeric data that the NetScaler appliance processes consists of dates and times. In addition to working with dates and times, the appliance processes other numeric data, such as the lengths of HTTP requests and responses. To process this data, you can configure default syntax expressions that process numbers.

A numeric expression consists of an expression prefix that returns a number and sometimes, but not always, an operator that can perform an operation on the number. Examples of expression prefixes that return numbers are SYS.TIME.DAY, HTTP.REQ.CONTENT_LENGTH, and HTTP.RES.BODY.LENGTH. Numeric operators can work with any prefix expression that returns data in numeric format. The GT(<int>) operator, for example, can be used with any prefix expression, such as HTTP.REQ.CONTENT_LENGTH, that returns an integer. Numeric expression prefixes and operators are also covered in "Compound Operations for Numbers" and "Default Syntax Expressions: Parsing HTTP, TCP, and UDP Data."

Format of Dates and Times in an Expression

Nov 14, 2013

When configuring a default syntax expression in a policy that works with dates and times (for example, the NetScaler system time or a date in an SSL certificate), you specify a time format as follows:

GMT|LOCAL [<yyyy>] [<month>] [<d>] [<h>] [<m>] [<s>]

Where:

- <yyyy> is a four-digit year after GMT or LOCAL.
- <month> is a three-character abbreviation for the month, for example, Jan, Dec.
- <d> is a day of the week or an integer for the date.
  You cannot specify the day as Monday, Tuesday, and so on. You specify either an integer for a specific day of the month, or you specify a date as the first, second, third weekday of the month, and so on. Following are examples of specifying a day of the week:
  - Sun_1 is the first Sunday of the month.
  - Sun_3 is the third Sunday of the month.
  - Wed_3 is the third Wednesday of the month.
  - 30 is an example of an exact date in a month.
- <h> is the hour, for example, 10h.
- <s> is the number of seconds, for example, 30s.

The following example expression is true if the date is between 2008 Jan and 2009 Jan, based on GMT.

```
http.req.date.between(GMT 2008 Jan, GMT 2009 Jan)
```

The following example expression is true for March and all months that follow March in the calendar year, based on GMT:

```
sys.time.ge(GMT 2008 Mar)
```

When you specify a date and time, note that the format is case sensitive and must preserve the exact number of blank spaces between entries.

Note: In an expression that requires two time values, both must use GMT or both must use LOCAL. You cannot mix the two in an expression.
Note: Unlike when you use the SYS.TIME prefix in a default syntax expression, if you specify SYS.TIME in a rewrite action, the NetScaler returns a string in conventional date format (for example, Sun, 06 Nov 1994 08:49:37 GMT). For example, the following rewrite action replaces the http.res.date header with the NetScaler system time in a conventional date format: add rewrite action `sync_date replace http.res.date sys.time`

Expressions for the NetScaler System Time

Mar 20, 2012

The SYS.TIME expression prefix extracts the NetScaler system time. You can configure expressions that establish whether a particular event occurred at a particular time or within a particular time range according to the NetScaler system time.

The following table describes the expressions that you can create by using the SYS.TIME prefix.

Table 1. Expressions That Return NetScaler System Dates and Times

NetScaler Time Operation	Description
SYS.TIME.BETWEEN(<time1>, <time2>)	Returns a Boolean TRUE if the returned value is later than <time1> and earlier than <time2>. You format the <time1>, <time2> arguments as follows: • They must both be GMT or both LOCAL. • <time2> must be later than <time1>. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following: • sys.time.between(GMT 2004, GMT 2006) • sys.time.between(GMT 2004 Jan, GMT 2006 Nov) • sys.time.between(GMT 2004 Jan, GMT 2006) • sys.time.between(GMT 2005 May Sun_1, GMT 2005 May Sun_3) • sys.time.between(GMT 2005 May 1, GMT May 2005 1) • sys.time.between(LOCAL 2005 May 1, LOCAL May 2005 1)
SYS.TIME.DAY	Returns the current day of the month as a number from 1 through 31.
SYS.TIME.EQ(<time>)	Returns a Boolean TRUE if the current time is equal to the <time> argument. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following (evaluation results are shown in parentheses): • sys.time.eq(GMT 2005) (TRUE in this example.) • sys.time.eq(GMT 2005 Dec) (FALSE in this example.) • sys.time.eq(LOCAL 2005 May) (Evaluates to TRUE or FALSE in this example, depending on the current time zone.) • sys.time.eq(GMT 10h) (TRUE in this example.) • sys.time.eq(GMT 10h 30s) (TRUE in this example.) • sys.time.eq(GMT May 10h) (TRUE in this example.) • sys.time.eq(GMT Sun) (TRUE in this example.) • sys.time.eq(GMT May Sun_1) (TRUE in this example.)

NetScaler Time Operation	Description
SYS.TIME.NE(<time>)	Returns a Boolean TRUE if the current time is not equal to the <time> argument.
SYS.TIME.GE(<time>)	Returns a Boolean TRUE if the current time is later than or equal to <time>. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following (evaluation results are shown in parentheses): • sys.time.ge(GMT 2004) (TRUE in this example.) • sys.time.ge(GMT 2005 Jan) (TRUE in this example.) • sys.time.ge(LOCAL 2005 May) (TRUE or FALSE in this example, depending on the current time zone.) • sys.time.ge(GMT 8h) (TRUE in this example.) • sys.time.ge(GMT 30m) (FALSE in this example.) • sys.time.ge(GMT May 10h) (TRUE in this example.) • sys.time.ge(GMT May 10h 0m) (TRUE in this example.) • sys.time.ge(GMT Sun) (TRUE in this example.) • sys.time.ge(GMT May Sun_1) (TRUE in this example.)
SYS.TIME.GT(<time>)	Returns a Boolean TRUE if the time value is later than the <time> argument. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following (evaluation results are shown in parentheses): • sys.time.gt(GMT 2004) (TRUE in this example.) • sys.time.gt(GMT 2005 Jan) (TRUE in this example.) • sys.time.gt(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • sys.time.gt(GMT 8h) (TRUE in this example.) • sys.time.gt(GMT 30m) (FALSE in this example.) • sys.time.gt(GMT May 10h) (FALSE in this example.) • sys.time.gt(GMT May 10h 0m) (TRUE in this example.) • sys.time.gt(GMT Sun) (FALSE in this example.) • sys.time.gt(GMT May Sun_1) (FALSE in this example.)
SYS.TIME.HOURS	Returns the current hour as an integer from 0 to 23.
SYS.TIME.LE(<time>)	Returns a Boolean TRUE if the current time value precedes or is equal to the <time> argument. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following (evaluation results are shown in parentheses):

NetScaler Time Operation	Description
	• sys.time.le(GMT 2006) (TRUE in this example.) • sys.time.le(GMT 2005 Dec) (TRUE in this example.) • sys.time.le(LOCAL 2005 May) (TRUE or FALSE depending on the current timezone.) • sys.time.le(GMT 8h) (FALSE in this example.) • sys.time.le(GMT 30m) (TRUE in this example.) • sys.time.le(GMT May 10h) (TRUE in this example.) • sys.time.le(GMT Jun 11h) (TRUE in this example.) • sys.time.le(GMT Wed) (TRUE in this example.) • sys.time.le(GMT May Sun_1) (TRUE in this example.)
SYS.TIME.LT(<time>)	Returns a Boolean TRUE if the current time value precedes the <time> argument. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following (evaluation results are shown in parentheses): • sys.time.lt(GMT 2006) (TRUE in this example.) • sys.time.lt.time.lt(GMT 2005 Dec) (TRUE in this example.) • sys.time.lt(LOCAL 2005 May) (TRUE or FALSE depending on the current time zone.) • sys.time.lt(GMT 8h) (FALSE in this example.) • sys.time.lt(GMT 30m) (TRUE in this example.) • sys.time.lt(GMT May 10h) (FALSE in this example.) • sys.time.lt(GMT Jun 11h) (TRUE in this example.) • sys.time.lt(GMT Wed) (TRUE in this example.) • sys.time.lt(GMT May Sun_1) (FALSE in this example.)
SYS.TIME.MINUTES	Returns the current minute as an integer from 0 to 59.
SYS.TIME.MONTH	Extracts the current month and returns an integer from 1 (January) to 12 (December).
SYS.TIME.RELATIVE_BOOT	Calculates the number of seconds to the closest previous or scheduled reboot, and returns an integer. If the closest boot time is in the past, the integer is negative. If it is in the future, the integer is positive.
SYS.TIME.RELATIVE_NOW	Calculates the number of seconds between the current NetScaler system time and the specified time, and returns an integer showing the difference. If the designated time is in the past, the integer is negative; if it is in the future, the integer is positive.

NetScaler Time Operation	Description
SYS.TIME.SECONDS	Extracts the seconds from the current NetScaler system time, and returns that value as an integer from 0 to 59.
SYS.TIME.WEEKDAY	Returns the current weekday as a value from 0 (Sunday) to 6 (Saturday).
SYS.TIME.WITHIN (<time1>, <time2>)	If you omit an element of time in <time1>, for example, the day or hour, it is assumed to have the lowest value in its range. If you omit an element in <time2>, it is assumed to have the highest value of its range. The ranges for the elements of time are as follows: month 1-12, day 1-31, weekday 0-6, hour 0-23, minutes 0-59 and seconds 0-59. If you specify the year, you must do so in both <time1> and <time2>. For example, if the time is GMT 2005 May 10 10h 15m 30s, and it is the second Tuesday of the month, you can specify the following (evaluation results are shown in parentheses): • sys.time.within(GMT 2004, GMT 2006) (TRUE in this example.) • sys.time.within(GMT 2004 Jan, GMT 2006 Mar) (FALSE, May is not in the range of January to March.) • sys.time.within(GMT Feb, GMT) (TRUE, May is in the range of February to December.) • sys.time.within(GMT Sun_1, GMT Sun_3) (TRUE, the second Tuesday is between the first Sunday and the third Sunday.) • sys.time.within(GMT 2005 May 1 10h, GMT May 2005 1 17h) (TRUE in this example.) • sys.time.within(LOCAL 2005 May 1, LOCAL May 2005 1) (TRUE or FALSE, depending on the NetScaler system time zone.)
SYS.TIME.YEAR	Extracts the year from the current system time and returns that value as a four-digit integer.

Expressions for SSL Certificate Dates

May 11, 2012

You can determine the validity period for SSL certificates by configuring an expression that contains the following prefix:

CLIENT.SSL.CLIENT_CERT

The following example expression matches a particular time for expiration with the information in the certificate:

client.ssl.client_cert.valid_not_after.eq(GMT 2009)

The following table describes time-based operations on SSL certificates. To obtain the expression you want, replace *certificate* in the expression in the first column with the prefix expression, "CLIENT.SSL.CLIENT_CERT".

Table 1. Operations on Certificate (client.ssl.client_cert) Dates and Times

SSL Certificate Operation	Description
<certificate>.VALID_NOT_AFTER	Returns the last day before certificate expiration. The return format is the number of seconds since GMT January 1, 1970 (0 hours, 0 minutes, 0 seconds).
<certificate>.VALID_NOT_AFTER.BETWEEN(<time1>, <time2>)	Returns a Boolean TRUE value if the certificate validity is between the <time1> and <time2> arguments. Both <time1> and <time2> must be fully specified. Following are examples: GMT 1995 Jan is fully specified. GMT Jan is not fully specified GMT 1995 20 is not fully specified. GMT Jan Mon_2 is not fully specified. The <time1> and <time2> arguments must be both GMT or both LOCAL, and <time2> must be greater than <time1>. For example, if it is GMT 2005 May 1 10h 15m 30s, and the first Sunday of the month, you can specify the following (evaluation results are in parentheses). • . . .between(GMT 2004, GMT 2006) (TRUE) • . . .between(GMT 2004 Jan, GMT 2006 Nov) (TRUE) • . . .between(GMT 2004 Jan, GMT 2006) (TRUE) • . . .between(GMT 2005 May Sun_1, GMT

SSL Certificate Operation	Description
	2005 May Sun_3) (TRUE) • . . .between(GMT 2005 May 1, GMT May 2005 1) (TRUE) • . . .between(LOCAL 2005 May 1, LOCAL May 2005 1) (TRUE or FALSE, depending on the NetScaler system time zone.)
<certificate>.VALID_NOT_AFTER.DAY	Extracts the last day of the month that the certificate is valid, and returns a number from 1 through 31, as appropriate for the date.
<certificate>.VALID_NOT_AFTER.EQ(<time>)	Returns a Boolean TRUE if the time is equal to the <time> argument. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following (evaluation results for this example are in parentheses): • . . .eq(GMT 2005) (TRUE) • . . .eq(GMT 2005 Dec) (FALSE) • . . .eq(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone) • . . .eq(GMT 10h) (TRUE) • . . .eq(GMT 10h 30s) (TRUE) • . . .eq(GMT May 10h) (TRUE) • . . .eq(GMT Sun) (TRUE) • . . .eq(GMT May Sun_1) (TRUE)
<certificate>.VALID_NOT_AFTER.GE(<time>)	Returns a Boolean TRUE if the time value is greater than or equal to the argument <time>. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .ge(GMT 2004) (TRUE) • . . .ge(GMT 2005 Jan) (TRUE) • . . .ge(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .ge(GMT 8h) (TRUE) • . . .ge(GMT 30m) (FALSE) • . . .ge(GMT May 10h) (TRUE) • . . .ge(GMT May 10h 0m) (TRUE)

SSL Certificate Operation	Description
	• . . .ge(GMT Sun) (TRUE) • . . .ge(GMT May Sun_1) (TRUE)
<certificate>.VALID_NOT_AFTER.GT(<time>)	Returns a Boolean TRUE if the time value is greater than the argument <time>. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .gt(GMT 2004) (TRUE) • . . .gt(GMT 2005 Jan) (TRUE) • . . .gt(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .gt(GMT 8h) (TRUE) • . . .gt(GMT 30m) (FALSE) • . . .gt(GMT May 10h) (FALSE) • . . .gt(GMT Sun) (FALSE) • . . .gt(GMT May Sun_1) (FALSE)
<certificate>.VALID_NOT_AFTER.HOURS	Extracts the last hour that the certificate is valid and returns that value as an integer from 0 to 23.
<certificate>.VALID_NOT_AFTER.LE(<time>)	Returns a Boolean TRUE if the time precedes or is equal to the <time> argument. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .le(GMT 2006) (TRUE) • . . .le(GMT 2005 Dec) (TRUE) • . . .le(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .le(GMT 8h) (FALSE) • . . .le(GMT 30m) (TRUE) • . . .le(GMT May 10h) (TRUE) • . . .le(GMT Jun 11h) (TRUE) • . . .le(GMT Wed) (TRUE) • . . .le(GMT May Sun_1) (TRUE)
<certificate>.VALID_NOT_AFTER.LT(<time>)	Returns a Boolean TRUE if the time precedes the

SSL Certificate Operation	Description
	<time> argument. For example, if the current time is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month, you can specify the following: • . . .lt(GMT 2006) (TRUE) • . . .lt(GMT 2005 Dec) (TRUE) • . . .lt(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .lt(GMT 8h) (FALSE) • . . .lt(GMT 30m) (TRUE) • . . .lt(GMT May 10h) (FALSE) • . . .lt(GMT Jun 11h) (TRUE) • . . .lt(GMT Wed) (TRUE) • . . .lt(GMT May Sun_1) (FALSE)
<certificate>.VALID_NOT_AFTER.MINUTES	Extracts the last minute that the certificate is valid and returns that value as an integer from 0 to 59.
<certificate>.VALID_NOT_AFTER.MONTH	Extracts the last month that the certificate is valid and returns that value as an integer from 1 (January) to 12 (December).
<certificate>.VALID_NOT_AFTER.RELATIVE_BOOT	Calculates the number of seconds to the closest previous or scheduled reboot and returns an integer. If the closest boot time is in the past, the integer is negative. If it is in the future, the integer is positive.
<certificate>.VALID_NOT_AFTER.RELATIVE_NOW	Calculates the number of seconds between the current system time and the specified time and returns an integer. If the time is in the past, the integer is negative; if it is in the future, the integer is positive.
<certificate>.VALID_NOT_AFTER.SECONDS	Extracts the last second that the certificate is valid and returns that value as an integer from 0 to 59.
<certificate>.VALID_NOT_AFTER.WEEKDAY	Extracts the last weekday that the certificate is valid. Returns a number between 0 (Sunday) and 6 (Saturday) to give the weekday in the time value.
<certificate>.VALID_NOT_AFTER.WITHIN(<time1>, <time2>)	Returns a Boolean TRUE if the time lies within all the ranges defined by the elements in <time1> and <time2>.

SSL Certificate Operation	Description
	If you omit an element of time from <time1>, it is assumed to have the lowest value in its range. If you omit an element from <time2>, it is assumed to have the highest value of its range. If you specify a year in <time1>, you must specify it in <time2>. The ranges for elements of time are as follows: month 1-12, day 1-31, weekday 0-6, hour 0-23, minutes 0-59 and seconds 0-59. For the result to be TRUE, each element in the time must exist in the corresponding range that you specify in <time1>, <time2>. For example, if time is GMT 2005 May 10 10h 15m 30s, and it is the second Tuesday of the month, you can specify the following (evaluation results are in parentheses): • . . .within(GMT 2004, GMT 2006) (TRUE) • . . .within(GMT 2004 Jan, GMT 2006 Mar) (FALSE, May is not in the range of January to March.) • . . .within(GMT Feb, GMT) (TRUE, May is in the range for February to December) • . . .within(GMT Sun_1, GMT Sun_3) (TRUE, the second Tuesday lies within the range of the first Sunday through the third Sunday) • . . .within(GMT 2005 May 1 10h, GMT May 2005 1 17h) (TRUE) • . . .within(LOCAL 2005 May 1, LOCAL May 2005 1) (TRUE or FALSE, depending on the NetScaler system time zone)
<certificate>.VALID_NOT_AFTER.YEAR	Extracts the last year that the certificate is valid and returns a four-digit integer.
<certificate>.VALID_NOT_BEFORE	Returns the date that the client certificate becomes valid. The return format is the number of seconds since GMT January 1, 1970 (0 hours, 0 minutes, 0 seconds).
<certificate>.VALID_NOT_BEFORE.BETWEEN(<time1>, <time2>)	Returns a Boolean TRUE if the time value is between the two time arguments. Both <time1> and <time2> arguments must be fully specified.

SSL Certificate Operation	Description
	Following are examples: • GMT 1995 Jan is fully specified. • GMT Jan is not fully specified. • GMT 1995 20 is not fully specified. • GMT Jan Mon_2 is not fully specified. The time arguments must be both GMT or both LOCAL, and <time2> must be greater than <time1>. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .between(GMT 2004, GMT 2006) (TRUE) • . . .between(GMT 2004 Jan, GMT 2006 Nov) (TRUE) • . . .between(GMT 2004 Jan, GMT 2006) (TRUE) • . . .between(GMT 2005 May Sun_1, GMT 2005 May Sun_3) (TRUE) • . . .between(GMT 2005 May 1, GMT May 2005 1) (TRUE) • . . .between(LOCAL 2005 May 1, LOCAL May 2005 1) (TRUE or FALSE, depending on the NetScaler system time zone.)
<certificate>.VALID_NOT_BEFORE.DAY	Extracts the last day of the month that the certificate is valid and returns that value as a number from 1 through 31 representing that day.
<certificate>.VALID_NOT_BEFORE.EQ(<time>)	Returns a Boolean TRUE if the time is equal to the <time> argument. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .eq(GMT 2005) (TRUE) • . . .eq(GMT 2005 Dec) (FALSE) • . . .eq(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .eq(GMT 10h) (TRUE) • . . .eq(GMT 10h 30s) (TRUE)

SSL Certificate Operation	Description
	• . . .eq(GMT May 10h) (TRUE) • . . .eq(GMT Sun) (TRUE) • . . .eq(GMT May Sun_1) (TRUE)
<certificate>.VALID_NOT_BEFORE.GE(<time>)	Returns a Boolean TRUE if the time is greater than (after) or equal to the <time> argument. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results are in parentheses): • . . .ge(GMT 2004) (TRUE) • . . .ge(GMT 2005 Jan) (TRUE) • . . .ge(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .ge(GMT 8h) (TRUE) • . . .ge(GMT 30m) (FALSE) • . . .ge(GMT May 10h) (TRUE) • . . .ge(GMT May 10h 0m) (TRUE) • . . .ge(GMT Sun) (TRUE) • . . .ge(GMT May Sun_1) (TRUE)
<certificate>.VALID_NOT_BEFORE.GT(<time>)	Returns a Boolean TRUE if the time occurs after the <time> argument. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results are in parentheses): • . . .gt(GMT 2004) (TRUE) • . . .gt(GMT 2005 Jan) (TRUE) • . . .gt(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .gt(GMT 8h) (TRUE) • . . .gt(GMT 30m) (FALSE) • . . .gt(GMT May 10h) (FALSE) • . . .gt(GMT May 10h 0m) (TRUE) • . . .gt(GMT Sun) (FALSE) • . . .gt(GMT May Sun_1) (FALSE)
<certificate>.VALID_NOT_BEFORE.HOURS	Extracts the last hour that the certificate is valid and returns that value as an integer from 0 to 23.
<certificate>.VALID_NOT_BEFORE.LE(<time>)	Returns a Boolean TRUE if the time precedes or is

SSL Certificate Operation	Description
	equal to the <time> argument. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .le(GMT 2006) (TRUE) • . . .le(GMT 2005 Dec) (TRUE) • . . .le(LOCAL 2005 May) (TRUE or FALSE,depending on the current time zone.) • . . .le(GMT 8h) (FALSE) • . . .le(GMT 30m) (TRUE) • . . .le(GMT May 10h) (TRUE) • . . .le(GMT Jun 11h) (TRUE) • . . .le(GMT Wed) (TRUE) • . . .le(GMT May Sun_1) (TRUE)
<certificate>.VALID_NOT_BEFORE.LT(<time>)	Returns a Boolean TRUE if the time precedes the <time> argument. For example, if the time value is GMT 2005 May 1 10h 15m 30s, and it is the first Sunday of the month of May in 2005, you can specify the following (evaluation results for this example are in parentheses): • . . .lt(GMT 2006) (TRUE) • . . .lt(GMT 2005 Dec) (TRUE) • . . .lt(LOCAL 2005 May) (TRUE or FALSE, depending on the current time zone.) • . . .lt(GMT 8h) (FALSE) • . . .lt(GMT 30m) (TRUE) • . . .lt(GMT May 10h) (FALSE) • . . .lt(GMT Jun 11h) (TRUE) • . . .lt(GMT Wed) (TRUE) • . . .lt(GMT May Sun_1) (FALSE)
<certificate>.VALID_NOT_BEFORE.MINUTES	Extracts the last minute that the certificate is valid. Returns the current minute as an integer from 0 to 59.
<certificate>.VALID_NOT_BEFORE.MONTH	Extracts the last month that the certificate is valid. Returns the current month as an integer from 1

SSL Certificate Operation	Description
<certificate>.VALID_NOT_BEFORE.RELATIVE_BOOT	Calculates the number of seconds to the closest previous or scheduled NetScaler reboot and returns an integer. If the closest boot time is in the past, the integer is negative; if it is in the future, the integer is positive.
<certificate>.VALID_NOT_BEFORE.RELATIVE_NOW	Returns the number of seconds between the current NetScaler system time and the specified time as an integer. If the designated time is in the past, the integer is negative. If it is in the future, the integer is positive.
<certificate>.VALID_NOT_BEFORE.SECONDS	Extracts the last second that the certificate is valid. Returns the current second as an integer from 0 to 59.
<certificate>.VALID_NOT_BEFORE.WEEKDAY	Extracts the last weekday that the certificate is valid. Returns the weekday as a number between 0 (Sunday) and 6 (Saturday).
<certificate>.VALID_NOT_BEFORE.WITHIN(<time1>, <time2>)	Returns a Boolean TRUE if each element of time exists within the range defined in the <time1>, <time2> arguments. If you omit an element of time from <time1>, it is assumed to have the lowest value in its range. If you omit an element of time from <time2>, it is assumed to have the highest value in its range. If you specify a year in <time1>, it must be specified in <time2>. The ranges for elements of time are as follows: month 1-12, day 1-31, weekday 0-6, hour 0-23, minutes 0-59 and seconds 0-59. For example, if the time is GMT 2005 May 10 10h 15m 30s, and it is the second Tuesday of the month, you can specify the following (evaluation results are in parentheses): • . . .within(GMT 2004, GMT 2006) (TRUE) • . . .within(GMT 2004 Jan, GMT 2006 Mar) (FALSE, May is not in the range of January to March.) • . . .within(GMT Feb, GMT) (TRUE, May is in the range of February to December.)

(January) to 12 (December).

SSL Certificate Operation	Description
	. . .within(GMT Sun_1, GMT Sun_3) (TRUE, the second Tuesday is between the first Sunday and the third Sunday.) . . .within(GMT 2005 May 1 10h, GMT May 2005 1 17h) (TRUE) . . .within(LOCAL 2005 May 1, LOCAL May 2005 1) (TRUE or FALSE, depending on the NetScaler system time zone)
<certificate>.VALID_NOT_BEFORE.YEAR	Extracts the last year that the certificate is valid. Returns the current year as a four-digit integer.

Expressions for HTTP Request and Response Dates

Sep 02, 2013

The following expression prefixes return the contents of the HTTP Date header as text or as a date object. These values can be evaluated as follows:

- As a number. The numeric value of an HTTP Date header is returned in the form of the number of seconds since Jan 1 1970.
  For example, the expression http.req.date.mod(86400) returns the number of seconds since the beginning of the day. These values can be evaluated using the same operations as other non-date-related numeric data. For more information, see "Expression Prefixes for Numeric Data Other Than Date and Time."
- As an HTTP header. Date headers can be evaluated using the same operations as other HTTP headers.
  For more information, see "Default Syntax Expressions: Parsing HTTP, TCP, and UDP Data."
- As text. Date headers can be evaluated using the same operations as other strings.

For more information, see "Default Syntax Expressions: Evaluating Text."

Table 1. Prefixes That Evaluate HTTP Date Headers

Prefix	Description
HTTP.REQ.DATE	Returns the contents of the HTTP Date header as text or as a date object. The date formats recognized are: RFC822. Sun, 06 Jan 1980 08:49:37 GMT RFC850. Sunday, 06-Jan-80 09:49:37 GMT ASCTIME. Sun Jan 6 08:49:37 1980
HTTP.RES.DATE	Returns the contents of the HTTP Date header as text or as a date object. The date formats recognized are: RFC822. Sun, 06 Jan 1980 8:49:37 GMT RFC850. Sunday, 06-Jan-80 9:49:37 GMT ASCTIME. Sun Jan 6 08:49:37 1980

Generating the Day of the Week, as a String, in Short and Long Formats

Mar 20, 2012

The functions, WEEKDAY_STRING_SHORT and WEEKDAY_STRING, generate the day of the week, as a string, in short and long formats, respectively. The strings that are returned are always in English. The prefix used with these functions must return the day of the week in integer format and the acceptable range for the value returned by the prefix is 0-6. Therefore, you can use any prefix that returns an integer in the acceptable range. An UNDEF condition is raised if the returned value is not in this range or if memory allocation fails.

Following are the descriptions of the functions:

Table 1. Functions That Generate the Day of the Week, as a String, in Short and Long Formats

Function	Description
<prefix>.WEEKDAY_STRING_SHORT	Returns the day of the week in short format. The short form is always 3 characters long with an initial capital and the remaining characters in lower case. For example, SYS.TIME.WEEKDAY.WEEKDAY_STRING_SHORT returns Sun if the value returned by the WEEKDAY function is 0 and Sat if the value returned by the prefix is 6.
<prefix>.WEEKDAY_STRING	Returns the day of the week in long format. The long form always has an initial capital, with the remaining characters in lower case. For example, SYS.TIME.WEEKDAY.WEEKDAY_STRING returns Sunday if the value returned by the WEEKDAY function is 0 and Saturday if the value returned by the prefix is 6.

Expression Prefixes for Numeric Data Other Than Date and Time

Sep 02, 2013

In addition to configuring expressions that operate on time, you can configure expressions for the following types of numeric data:

- The length of HTTP requests, the number of HTTP headers in a request, and so on.
  For more information, see "Expressions for Numeric HTTP Payload Data Other Than Dates."
- IP and MAC addresses.
  For more information, see "Expressions for IP Addresses and IP Subnets."
- Client and server data in regard to interface IDs and transaction throughput rate.
  For more information, see "Expressions for Numeric Client and Server Data."
- Numeric data in client certificates other than dates.
  For information on these prefixes, including the number of days until certificate expiration and the encryption key size, see "Prefixes for Numeric Data in SSL Certificates."

Converting Numbers to Text

Sep 02, 2013

The following functions produce binary strings from a number returned by an expression prefix. These functions are particularly useful in the TCP rewrite feature as replacement strings for binary data. For more information about the TCP rewrite feature, see "Rewrite."

All the functions return a value of type **text**. The **endianness** that some of the functions accept as a parameter is either LITTLE_ENDIAN or BIG_ENDIAN.

Table 1. Functions That Produce a Binary String From a Number

Function	Description
<number>.SIGNED8_STRING	Produces an 8-bit signed binary string representing the number. If the value is out of range, an **undef** condition is raised **Example** HTTP.REQ.BODY(100).GET_SIGNED8(16).SUB(3).SIGNED8_STRING
<number>.UNSIGNED8_STRING	Produces an 8-bit unsigned binary string representing the number. If the value is out of range, an **undef** condition is rais **Example** HTTP.REQ.BODY(100).GET_UNSIGNED8(31).ADD(3).UNSIGNED8_STRING
<number>.SIGNED16_STRING(<endianness>)	Produces a 16-bit signed binary string representing the number. If the value is out of range, an **undef** condition is raised. **Example** HTTP.REQ.BODY(100).SKIP(12).GET_SIGNED16(0, BIG_ENDIAN).SUB(4).SIGNED16_STRING(BIG_ENDIAN
<number>.UNSIGNED16_STRING(<endianness>)	Produces a 16-bit unsigned binary string representing the number. If the value is out of range, an **undef** condition is rais **Example** HTTP.REQ.BODY(100).GET_UNSIGNED16(47, LITTLE_ENDIAN).ADD(7).UNSIGNED16_STRING(LITTLE_ENDIAN)
<number>.SIGNED32_STRING(<endianness>)	Produces a 32-bit signed binary string representing the number. **Example** HTTP.REQ.BODY(100).AFTER_STR("delim").GET_SIGNED32(0, BIG_ENDIAN).SUB(1).SIGNED32_STRING(BIG_ENDIAN)
<unsigned_long_number>.UNSIGNED8_STRING	Produces an 8-bit unsigned binary string representing the number. If the value is out of range, an **undef** condition is rais **Example** HTTP.REQ.BODY(100).GET_UNSIGNED8(24).TYPECAST_UNSIGNED_LONG_AT.ADD(12).UNSIGNED8_STRI
<unsigned_long_number>.UNSIGNED16_STRING(<endianness>)	Produces a 16-bit unsigned binary string representing the number. If the value is out of range, an **undef** condition is rais **Example** HTTP.REQ.BODY(100).GET_UNSIGNED16(23, LITTLE_ENDIAN).TYPECAST_UNSIGNED_LONG_AT.ADD(10).UNSIGNED16_STRING(LITTLE_ENDIAN)
<unsigned_long_number>.UNSIGNED32_STRING(<endianness>)	Produces a 32-bit unsigned binary string representing the number. If the value is out of range, an **undef** condition is rais **Example** HTTP.REQ.BODY(100).AFTER_STR("delim2").GET_UNSIGNED32(0, BIG_ENDIAN).ADD(2).UNSIGNED32_STRING(BIG_ENDIAN)

Virtual Server Based Expressions

Apr 17, 2012

The SYS.VSERVER("<vserver-name>") expression prefix enables you to identify a virtual server. You can use the following functions with this prefix to retrieve information related to the specified virtual server:

- **THROUGHPUT**. Returns the throughput of the virtual server in Mbps (Megabits per second). The value returned is an unsigned long number.
  Usage: SYS.VSERVER("vserver").THROUGHPUT
- **CONNECTIONS**. Returns the number of connections being managed by the virtual server. The value returned is an unsigned long number.
  Usage: SYS.VSERVER("vserver").CONNECTIONS
- **STATE**. Returns the state of the virtual server. The value returned is UP, DOWN, or OUT_OF_SERVICE. One of these values can therefore be passed as an argument to the EQ() operator to perform a comparison that results in a Boolean TRUE or FALSE.
  Usage: SYS.VSERVER("vserver").STATE
- **HEALTH**. Returns the percentage of services in an UP state for the specified virtual server. The value returned is an integer.
  Usage: SYS.VSERVER("vserver").HEALTH
- **RESPTIME**. Returns the response time as an integer representing the number of microseconds. Response time is the average TTFB (Time To First Byte) from all the services bound to the virtual server.
  Usage: SYS.VSERVER("vserver").RESPTIME
- **SURGECOUNT**. Returns the number of requests in the surge queue of the virtual server. The value returned is an integer.
  Usage: SYS.VSERVER("vserver").SURGECOUNT

Example 1

The following rewrite policy aborts rewrite processing if the number of connections at the load balancing virtual server LBvserver exceeds 10000:

```
add rewrite policy norewrite_pol sys.vserver("LBvserver").connections.gt(10000) norewrite
```

Example 2

The following rewrite action inserts a custom header, TP, whose value is the throughout at the virtual server LBvserver:

```
add rewrite action tp_header insert_http_header TP SYS.VSERVER("LBvserver").THROUGHPUT
```

Example 3

The following audit log message action writes the average TTFB of the services bound to a virtual server, to the newnslog log file:

```
add audit messageaction log_vserver_resptime_act INFORMATIONAL "\"NS Response Time to Servers:\" +
sys.vserver(\"ssllb\").resptime + \" millisec\"" -logtoNewnslog YES -bypassSafetyCheck YES
```

Default Syntax Expressions: Parsing HTTP, TCP, and UDP Data

Sep 02, 2013

You can configure default syntax expressions to evaluate and process the payload in HTTP requests and responses. The payload associated with an HTTP connection includes the various HTTP headers (both standard and custom headers), the body, and other connection information such as the URL. Additionally, you can evaluate and process the payload in a TCP or UDP packet. For HTTP connections, for example, you can check whether a particular HTTP header is present or if the URL includes a particular query parameter.

You can configure expressions to transform the URL encoding and apply HTML or XML “safe” coding for subsequent evaluation. You can also use XPATH and JSON prefixes to evaluate date in XML and JSON files, respectively.

You can also use text-based and numeric default syntax expressions to evaluate HTTP request and response data. For more information, see "Default Syntax Expressions: Evaluating Text" and "Default Syntax Expressions: Working with Dates, Times, and Numbers."

About Evaluating HTTP and TCP Payload

Sep 02, 2013

The payload of an HTTP request or response consists of HTTP protocol information such as headers, a URL, body content, and version and status information. When you configure a default syntax expression to evaluate HTTP payload, you use a default syntax expression prefix and, if necessary, an operator.

For example, you use the following expression, which includes the http.req.header("<header_name>") prefix and the exists operator, if you want to determine whether an HTTP connection includes a custom header named "myHeader":

```
http.req.header("myHeader").exists
```

You can also combine multiple default syntax expressions with Boolean and arithmetic operators. For example, the following compound expression could be useful with various NetScaler features, such as Integrated Caching, Rewrite, and Responder. This expression first uses the && Boolean operator to determine whether an HTTP connection includes the Content-Type header with a value of "text/html." If that operation returns a value of FALSE, the expression determines whether the HTTP connection includes a "Transfer-Encoding" or "Content-Length" header.

```
(http.req.header("Content-Type").exists && http.req.header("Content-Type").eq("text/html")) ||
(http.req.header("Transfer-Encoding").exists) || (http.req.header("Content-Length").exists)
```

The payload of a TCP or UDP packet is the data portion of the packet. You can configure default syntax expressions to examine features of a TCP or UDP packet, including the following:

- Source and destination domains
- Source and destination ports
- The text in the payload
- Record types

The following expression prefixes extract text from the body of the payload:

- HTTP.REQ.BODY(integer). Returns the body of an HTTP request as a multiline text object, up to the character position designated in the integer argument. If there are fewer characters in the body than is specified in the argument, the entire body is returned.
- HTTP.RES.BODY(integer). Returns a portion of the HTTP response body. The length of the returned text is equal to the number in the integer argument. If there are fewer characters in the body than is specified in integer, the entire body is returned.
- CLIENT.TCP.PAYLOAD(integer). Returns TCP payload data as a string, starting with the first character in the payload and continuing for the number of characters in the integer argument.

Following is an example that evaluates to TRUE if a response body of 1024 bytes contains the string "https", and this string occurs after the string "start string" and before the string "end string":

```
http.res.body(1024).after_str("start_string").before_str("end_string").contains("https")
```

Note: You can apply any text operation to the payload body. For information on operations that you can apply to text, see "Default Syntax Expressions: Evaluating Text."

Expressions for Identifying the Protocol in an Incoming IP Packet

May 21, 2015

The following table lists the expressions that you can use to identify the protocol in an incoming packet.

Expression	Description
CLIENT.IP.PROTOCOL	Identifies the protocol in IPv4 packets sent by clients.
CLIENT.IPV6.PROTOCOL	Identifies the protocol in IPv6 packets sent by clients.
SERVER.IP.PROTOCOL	Identifies the protocol in IPv4 packets sent by servers.
SERVER.IPV6.PROTOCOL	Identifies the protocol in IPv6 packets sent by servers.

You can pass the Internet Assigned Numbers Authority (IANA) protocol number to the PROTOCOL function. For example, if you want to determine whether the protocol in an incoming packet is TCP, you can use CLIENT.IP.PROTOCOL.EQ(6), where 6 is the IANA-assigned protocol number for TCP. For some protocols, you can pass an enumeration value instead of the protocol number. For example, instead of CLIENT.IP.PROTOCOL.EQ(6), you can use CLIENT.IP.PROTOCOL.EQ(TCP). The following table lists the protocols for which you can use enumeration values, and the corresponding enumeration values for use with the PROTOCOL function.

Protocol	Enumeration value
Transmission Control Protocol (TCP)	TCP
User Datagram Protocol (UDP)	UDP
Internet Control Message Protocol (ICMP)	ICMP
IP Authentication Header (AH), for providing authentication services in IPv4 and IPv6	AH
Encapsulating Security Payload (ESP) protocol	ESP
General Routing Encapsulation (GRE)	GRE
IP-within-IP Encapsulation Protocol	IPIP

Protocol	Expression value
Internet Control Message Protocol for IPv6 (ICMPv6)	ICMPV6
Fragment Header for IPv6	FRAGMENT

The protocol expressions can be used in both request-based and response-based policies. You can use the expressions in various NetScaler features, such as load balancing, WAN optimization, content switching, rewrite, and listen policies. You can use the expressions with functions such as EQ() and NE(), to identify the protocol in a policy and perform an action.

Following are some use cases for the expressions:

- In Branch Repeater load balancing configurations, you can use the expressions in a listen policy for the wildcard virtual server. For example, you can configure the wildcard virtual server with the listen policy CLIENT.IP.PROTOCOL.EQ(TCP) so that the virtual server processes only TCP traffic and simply bridges all non-TCP traffic. Even though you can use an Access Control List instead of the listen policy, the listen policy provides better control over what traffic is processed.
- For content switching virtual servers of type ANY, you can configure content switching policies that switch requests on the basis of the protocol in incoming packets. For example, you can configure content switching policies to direct all TCP traffic to one load balancing virtual server and all non-TCP traffic to another load balancing virtual server.
- You can use the client-based expressions to configure persistence based on the protocol. For example, you can use CLIENT.IP.PROTOCOL to configure persistence on the basis of the protocols in incoming IPv4 packets.

Expressions for HTTP and Cache-Control Headers

Sep 02, 2013

One common method of evaluating HTTP traffic is to examine the headers in a request or a response. A header can perform a number of functions, including the following:

- Provide cookies that contain data about the sender.
- Identify the type of data that is being transmitted.
- Identify the route that the data has traveled (the Via header).

Note: Note that if an operation is used to evaluate both header and text data, the header-based operation always overrides the text-based operation. For example, the AFTER_STR operation, when applied to a header, overrides text-based AFTER_STR operations for all instances of the current header type.

The following table describes expression prefixes that extract HTTP headers.

Table 1. Prefixes That Extract HTTP Headers

HTTP Header Prefix	Description
HTTP.REQ.HEADER("<header_name>")	Returns the contents of the HTTP header specified by the <header_name> argument. The header name cannot exceed 32 characters. Note that this prefix returns the value from the Host header by default. To use this value as a host name you need to typecast it as follows: http.req.header("host").typecast_http_hostname_t For more information on typecasting, see "Typecasting Data."
HTTP.REQ.FULL_HEADER	Returns the contents of the complete set of HTTP header fields including the request line (for example, "GET /brochures/index.html HTTP/1.1") and the terminating \r\n\r\n sequence.
HTTP.REQ.DATE	Returns the contents of the HTTP Date header. The following date formats are recognized: RFC822. Sun, 06 Jan 1980 8:49:37 GMT RFC850. Sunday, 06-Jan-80 9:49:37 GMT ASCII TIME. Sun Jan 6 08:49:37 1980 To evaluate a Date header as a date object, see "Default Syntax Expressions: Working with Dates, Times, and Numbers."
HTTP.REQ.COOKIE	(Name/Value List) Returns the contents of the HTTP Cookie header.
HTTP.REQ.TXID	Returns the HTTP transaction ID. The value is a function of an internal transaction number, system boot time and system MAC address.
HTTP.RES.HEADER("<header_name>")	Returns the contents of the HTTP header specified by the <header_name> argument. The header name cannot exceed 32 characters.
HTTP.RES.FULL_HEADER	Returns the contents of the complete set of HTTP header fields including the status line (for example, "HTTP/1.1 200 OK") and the terminating \r\n\r\n sequence.
HTTP.RES.SET_COOKIE	Returns the HTTP Set-Cookie header object in a response.

HTTP Header Prefix	Description
or HTTP.RES.SET_COOKIE2	
HTTP.RES.SET_COOKIE("<name>") or HTTP.RES.SET_COOKIE2("<name>")	Returns the cookie of the specified name if it is present. If it is not present, returns a text object of length 0. Returns UNDEF if more than 15 Set-Cookie headers are present and the specified cookie was not found in these headers.
HTTP.RES.SET_COOKIE("<name>").DOMAIN or HTTP.RES.SET_COOKIE2("<name>").DOMAIN	Returns the value of the first Domain field in the cookie. For example, if the cookie is Set-Cookie : Customer = "ABC"; DOMAIN=".abc.com"; DOMAIN=.xyz.com, the following expression returns .abc.com: http.res.set_cookie.cookie("customer").domain A string of zero length is returned if the Domain field or its value is absent.
HTTP.RES.SET_COOKIE.EXISTS("<name>") or HTTP.RES.SET_COOKIE2.EXISTS("<name>")	Returns a Boolean TRUE if a Cookie with the name specified in the <name> argument exists in the Set-Cookie header. This prefix returns UNDEF if more than 15 Set-Cookie headers are present and the named cookie is not in the first 15 headers.
HTTP.RES.SET_COOKIE.COOKIE("<name>").EXPIRES or HTTP.RES.SET_COOKIE2.COOKIE("<name>").EXPIRES	Returns the Expires field of the cookie. This is a date string that can be evaluated as a number, as a time object, or as text. If multiple Expires fields are present, the first one is returned. If the Expires field is absent, a text object of length zero is returned. To evaluate the returned value as a time object, see "Default Syntax Expressions: Working with Dates, Times, and Numbers."
HTTP.RES.SET_COOKIE.COOKIE("<name>").PATH|PATH.GET(n) or HTTP.RES.SET_COOKIE2.COOKIE("<name>").PATH|PATH.GET(n)	Returns the value of Path field of the cookie as a slash- (“/”) separated list. Multiple instances of a slash are treated as single slash. If multiple Path fields are present, the value of the first instance is returned. For example, the following is a cookie with two path fields: Set-Cookie : Customer = "ABC"; PATH="/a//b/c"; PATH= "/x/y/z" The following expression returns /a//b/c from this cookie: http.res.set_cookie.cookie("Customer").path The following expression returns b: http.res.set_cookie.cookie("Customer").path.get(2) Quotes are stripped from the returned value. A string of zero length is returned if the Path field or its value is absent.
HTTP.RES.SET_COOKIE.COOKIE("<name>").PATH.IGNORE_EMPTY_ELEMENTS or HTTP.RES.SET_COOKIE2.COOKIE("<name>").PATH.IGNORE_EMPTY_ELEMENTS	Ignores the empty elements in the list. For example, in the list a=10,b=11, ,c=89, the element delimiter in the list is , and the list has an empty element following a=10. The element following b=11 is not considered an empty element. As another example, in the following expression, if a request contains Cust_Header : 123,,24, ,15 the following expression returns a value of 4: http.req.header("Cust_Header").typecast_list_t(',').ignore_empty_elements.count The following expression returns a value of 5:

HTTP Header Prefix	Description
	http.req.header("Cust_Header").typecast_list_t(',').count
HTTP.RES.SET_COOKIE.COOKIE("<name>").PORT or HTTP.RES.SET_COOKIE2.COOKIE("<name>").PORT	Returns the value of Port field of the cookie. Operate as a comma-separated list. For example, the following expression returns 80. 2580 from Set-Cookie : Customer = "ABC"; PATH="/a/b/c"; PORT= "80, 2580": http.res.set_cookie.cookie("ABC").port A string of zero length is returned if the Port field or value is absent.
HTTP.RES.SET_COOKIE.COOKIE("<name>").PORT.IGNORE_EMPTY_ELEMENTS or HTTP.RES.SET_COOKIE2.COOKIE("<name>").PORT.IGNORE_EMPTY_ELEMENTS	Ignores the empty elements in the list. For example, in the list a=10,b=11, ,c=89, the element delimiter in the list is , and the list has an empty element following a=10. The element following b=11 is not considered an empty element. As another example, in the following expression, if a request contains Cust_Header : 123,,24, ,15 the following expression returns a value of 4: http.req.header("Cust_Header").typecast_list_t(',').ignore_empty_elements.count The following expression returns a value of 5: http.req.header("Cust_Header").typecast_list_t(',').count
HTTP.RES.SET_COOKIE.COOKIE("<name>").VERSION or HTTP.RES.SET_COOKIE2.COOKIE("<name>").VERSION	Returns the value of the first Version field in the cookie as a decimal integer. For example, the following expression returns 1 from the cookie Set-Cookie : Customer = "ABC"; VERSION = "1"; VERSION = "0" http.res.set_cookie.cookie("CUSTOMER").version A zero is returned if the Version field or its value is absent or if the value is not a decimal number.
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>) or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>)	Returns the nth instance (0-based) of the cookie with the specified name. If the cookie is absent, returns a text object of length 0. Returns UNDEF if more than 15 Set-Cookie headers are present and the cookie is not found.
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).DOMAIN or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).DOMAIN	Returns the value of the Domain field of the first cookie with the specified name. For example, the following expression returns a value of abc.com from the cookie Set-Cookie : Customer = "ABC"; DOMAIN=".abc.com"; DOMAIN=.xyz.com http.res.set_cookie.cookie("CUSTOMER").domain A string of zero length is returned if the Domain field or its value is absent.
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).EXPIRES or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).EXPIRES	Returns the *n*th instance (0-based) of the Expires field of the cookie with the specified name as a date string. The value can be operated upon as a time object that supports a number of date formats. If the Expires attribute is absent a string of length zero is returned.
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).PATH | PATH.GET(i)	Returns the value of the Path field of the nth cookie, as a '/' separated list. Multiple /s are treated as a single /.

HTTP Header Prefix	Description
or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).PATH | PATH.GET(i)	For example, the following expression returns /a//b/c from the cookie Set-Cookie : Customer = "ABC"; PATH="/a//b/c"; PATH= "/x/y/z" http.res.set_cookie.cookie("CUSTOMER").path The following returns b: http.res.set_cookie.cookie("CUSTOMER").path.get(2) A string of zero length is returned if the Path field or its value is absent.
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).PATH.IGNORE_EMPTY_ELEMENTS or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).PATH.IGNORE_EMPTY_ELEMENTS	Ignores the empty elements in the list. For example, in the list a=10,b=11, ,c=89, the element delimiter in the list is , and the list has an empty element following a=10. The element following b=11 is not considered an empty element. As another example, in the following expression, if a request contains Cust_Header : 123,,24, ,15 the following expression returns a value of 4: http.req.header("Cust_Header").typecast_list_t(',').ignore_empty_elements.count The following expression returns a value of 5: http.req.header("Cust_Header").typecast_list_t(',').count
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).PORT or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).PORT	Returns the value or values of the Port field of the named cookie as a ',' separated list. For example, the following expression returns 80, 2580 from the cookie Set-Cookie : Customer = "ABC"; PATH="/a/b/c"; PORT= "80, 2580" http.res.set_cookie.cookie("ABC").port A string of zero length is returned if the Port field or its value is absent.
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).PORT.IGNORE_EMPTY_ELEMENTS or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).PORT.IGNORE_EMPTY_ELEMENTS	Ignores the empty elements in the list. For example, in the list a=10,b=11, ,c=89, the element delimiter in the list is , and the list has an empty element following a=10. The element following b=11 is not considered an empty element. As another example, in the following expression, if a request contains Cust_Header : 123,,24, ,15 the following expression returns a value of 4: http.req.header("Cust_Header").typecast_list_t(',').ignore_empty_elements.count The following expression returns a value of 5: http.req.header("Cust_Header").typecast_list_t(',').count
HTTP.RES.SET_COOKIE.COOKIE("<name>", <integer>).VERSION or HTTP.RES.SET_COOKIE2.COOKIE("<name>", <integer>).VERSION	Returns the value of Version field of the *n*th cookie as a decimal integer. A string of zero length is returned if the Port field or its value is absent.
HTTP.RES.TXID	Returns the HTTP transaction ID. The value is a function of an internal transaction number, system boot time and system MAC address.

The following table describes operations that you can specify with the prefixes for HTTP headers.

Table 2. Operations That Evaluate HTTP Headers

HTTP Header Operation	Description
http header .EXISTS	Returns a Boolean TRUE if an instance of the specified header type exists. Following is an example: http.req.header("Cache-Control").exists
http header.CONTAINS" http header . CONTAINS(" <string>")	Returns a Boolean TRUE if the <string> argument appears in any instance of the header value. Note: This operation overrides any text-based Contains operations on all instances of the current header type. Following is an example of request with two headers: HTTP/1.1 200 OK\r\n MyHeader: abc\r\n Content-Length: 200\r\n MyHeader: def\r\n \r\n The following returns a Boolean TRUE: http.res.header("MyHeader"). contains("de") The following returns FALSE. Note that the NetScaler does not concatenate the different values. http.res.header("MyHeader"). contains("bcd")
http header .COUNT	Returns the number of headers in a request or response, to a maximum of 15 headers of the same type. The result is undefined if there are more than 15 instances of the header. Following is sample data in a request: HTTP/1.1 200 OK\r\n MyHeader: abc\r\n Content-Length: 200\r\n MyHeader: def\r\n \r\n When evaluating the preceding request, the following returns a count of 2: http.res.header("MyHeader").count
http header.AFTER_STR(" <string>")	Extracts the text that follows the first occurrence of the <string> argument. The headers are evaluated from the last instance to the first. Following is an example of a request: HTTP/1.1 200 OK\r\n MyHeader: 111abc\r\n Content-Length: 200\r\n MyHeader: 111def\r\n

	\r\n The following extracts the string "def" from the last instance of MyHeader. This is value "111def." http.res.header("MyHeader").after_str("111") The following extracts the string "c" from the first instance of MyHeader. This is the value "abc111." http.res.header("MyHeader").after_str("1ab")
http header.BEFORE_STR("<string>")	Extracts the text that appears prior to the first occurrence of the input <string> argument. The headers are evaluated from the last instance to the first. Following is an example of a request that contains headers: HTTP/1.1 200 OK\r\n MyHeader: abc111\r\n Content-Length: 200\r\n MyHeader: def111\r\n \r\n The following extracts the string "def" from the last instance of MyHeader. This is the value "def111." http.res.header("MyHeader").before_str("111") The following extracts the string "a" from the first instance of MyHeader. This is the value "abc111." http.res.header("MyHeader").before_str("bc1")
http header.INSTANCE(<instance number>)	An HTTP header can occur multiple times in a request or a response. This operation returns the header that occurs <instance number> of places before the final instance. For example, instance(0) selects the last instance of the current type, instance(1) selects the next-to-last instance, and so on. This prefix cannot be used in bidirectional policies. The <instance number> argument cannot exceed 14. Following is an example of a request with two headers: HTTP/1.1 200 OK\r\n MyHeader: abc\r\n Content-Length: 200\r\n MyHeader: def\r\n \r\n The following returns a text object that refers to "MyHeader: abc\r\n": http.res.header("MyHeader").instance(1)
http header.SUBSTR("<string>")	Extracts the text that matches the <string> argument. The headers are evaluated from the last instance to the first. Following is an example of a request with two headers that contain the string “111”: HTTP/1.1 200 OK\r\n MyHeader: abc111\r\n Content-Length: 200\r\n MyHeader: 111def\r\n \r\n

	The following returns "111" from the last instance of MyHeader. This is the header with the value "111def." http.res.header("MyHeader").substr("111")
http header.VALUE(<instance number>)	An HTTP header can occur multiple times in a request or a response. VALUE(0) selects the value in the last instance, VALUE(1) selects the value in the next-to-last instance, and so on. The <instance number> argument cannot exceed 14. Following is an example of a request with two headers: HTTP/1.1 200 OK\r\n MyHeader: abc\r\n Content-Length: 200\r\n MyHeader: def\r\n \r\n The following returns "abc\r\n": http.res.header("MyHeader").value(1)

The following prefixes apply specifically to Cache-Control headers.

Table 3. Prefixes That Extract Cache-Control Headers

HTTP Header Prefix	Description
HTTP.REQ.CACHE_CONTROL	Returns a Cache-Control header in an HTTP request.
HTTP.RES.CACHE_CONTROL	Returns a Cache-Control header in an HTTP response.

You can apply any of the operations for HTTP headers to Cache-Control headers. For more information, see "Operations for HTTP Headers."

In addition, the following operations identify specific types of Cache-Control headers. See RFC 2616 for information about these header types.

Table 4. Operations That Evaluate Cache-Control Headers

HTTP Header Operation	Description
Cache-Control header.NAME(<integer>)	Returns as a text value the name of the Cache-Control header that corresponds to the nth component in a name-value list, as specified by <integer>. The index of the name-value component is 0-based. If the <integer> that is specified by the integer argument is greater than the number of components in the list, a zero-length text object is returned. Following is an example: http.req.cache_control.name(3).contains("some_text")
Cache-Control header.IS_INVALID	Returns a Boolean TRUE if the Cache-Control header is not present in the request or response. Following is an example: http.req.cache_control.is_invalid

HTTP Header Operation	Description
Cache-Control header.IS_PRIVATE	Returns a Boolean TRUE if the Cache-Control header has the value Private. Following is an example: http.req.cache_control.is_private
Cache-Control header.IS_PUBLIC	Returns a Boolean TRUE if the Cache-Control header has the value Private. Following is an example: http.req.cache_control.is_public
Cache-Control header.IS_NO_STORE	Returns a Boolean TRUE if the Cache-Control header has the value No-Store. Following is an example: http.req.cache_control.is_no_store
Cache-Control header.IS_NO_CACHE	Returns a Boolean TRUE if the Cache-Control header has the value No-Cache. Following is an example: http.req.cache_control.is_no_cache
Cache-Control header.IS_MAX_AGE	Returns a Boolean TRUE if the Cache-Control header has the value Max-Age. Following is an example: http.req.cache_control.is_max_age
Cache-Control header.IS_MIN_FRESH	Returns a Boolean TRUE if the Cache-Control header has the value Min-Fresh. Following is an example: http.req.cache_control.is_min_fresh
Cache-Control header.IS_MAX_STALE	Returns a Boolean TRUE if the Cache-Control header has the value Max-Stale. Following is an example: http.req.cache_control.is_max_stale
Cache-Control header.IS_MUST_REVALIDATE	Returns a Boolean TRUE if the Cache-Control header has the value Must-Revalidate. Following is an example: http.req.cache_control.is_must_revalidate
Cache-Control header.IS_NO_TRANSFORM	Returns a Boolean TRUE if the Cache-Control header has the value No-Transform. Following is an example: http.req.cache_control.is_no_transform
Cache-Control header.IS_ONLY_IF_CACHED	Returns a Boolean TRUE if the Cache-Control header has the value Only-If-Cached. Following is an example: http.req.cache_control.is_only_if_cached

HTTP Header Operation	Description
Cache-Control header.IS_PROXY_REVALIDATE	Returns a Boolean TRUE if the Cache-Control header has the value Proxy-Revalidate. Following is an example: http.req.cache_control.is_proxy_revalidate
Cache-Control header.IS_S_MAXAGE	Returns a Boolean TRUE if the Cache-Control header has the value S-Maxage. Following is an example: http.req.cache_control.is_s_maxage
Cache-Control header.IS_UNKNOWN	Returns a Boolean TRUE if the Cache-Control header is of an unknown type. Following is an example: http.req.cache_control.is_unknown
Cache-Control header.MAX_AGE	Returns the value of the Cache-Control header Max-Age. If this header is absent or invalid, 0 is returned. Following is an example: http.req.cache_control.max_age.le(3)
Cache-Control header.MAX_STALE	Returns the value of the Cache-Control header Max-Stale. If this header is absent or invalid, 0 is returned. Following is an example: http.req.cache_control.max_stale.le(3)
Cache-Control header.MIN_FRESH	Returns the value of the Cache-Control header Min-Fresh. If this header is absent or invalid, 0 is returned. Following is an example: http.req.cache_control.min_fresh.le(3)
Cache-Control header.S_MAXAGE	Returns the value of the Cache-Control header S-Maxage. If this header is absent or invalid, 0 is returned. Following is an example: http.req.cache_control.s_maxage.eq(2)

Expressions for Extracting Segments of URLs

Sep 02, 2013

You can extract URLs and portions of URLs, such as the host name, or a segment of the URL path. For example, the following expression identifies HTTP requests for image files by extracting image file suffixes from the URL:

```
http.req.url.suffix.eq("jpeg") || http.req.url.suffix.eq("gif")
```

Most expressions for URLs operate on text and are described in "Expression Prefixes for Text in HTTP Requests and Responses." This section discusses the GET operation. The GET operation extracts text when used with the following prefixes:

- HTTP.REQ.URL.PATH
- VPN.BASEURL.PATH
- VPN.CLIENTLESS_BASEURL.PATH

The following table describes prefixes for HTTP URLs.

Table 1. Prefixes That Extract URLs

URL Prefix	Description
HTTP.REQ.URL.PATH.GET(<n>)	Returns a slash- (“/”) separated list from the URL path. For example, consider the following URL: http://www.mycompany.com/dir1/dir2/dir3/index.html?a=1 The following expression returns dir1 from this URL: http.req.url.path.get(1) The following expression returns dir2: http.req.url.path.get(2)
HTTP.REQ.URL.PATH.GET_REVERSE(<n>)	Returns a slash- (“/”) separated list from the URL path, starting from the end of the path. For example, consider the following URL: http://www.mycompany.com/dir1/dir2/dir3/index.html?a=1 The following expression returns index.html from this URL: http.req.url.path.get_reverse(0) The following expression returns dir3: http.req.url.path.get_reverse(1)

Expressions for HTTP Status Codes and Numeric HTTP Payload Data Other Than Dates

Mar 20, 2012

The following table describes prefixes for numeric values in HTTP data other than dates.

Table 1. Prefixes That Evaluate HTTP Request or Response Length

Prefix	Description
HTTP.REQ.CONTENT_LENGTH	Returns the length of an HTTP request as a number. Following is an example: `http.req.content_length < 500`
HTTP.RES.CONTENT_LENGTH	Returns the length of the HTTP response as a number. Following is an example: `http.res.content_length <= 1000`
HTTP.RES.STATUS	Returns the response status code

HTTP.RES.IS_REDIRECT	Returns a Boolean TRUE if the response code is associated with a redirect. Following are the redirect response codes:
	• 300 (Multiple Choices)
	• 301 (Moved Permanently)
	• 302 (Found)
	• 303 (See Other)
	• 305 (Use Proxy)
	• 307 (Temporary Redirect)
	Note: Status code 304 is not considered a redirect HTTP response status code. Status code 306 is unused.
	In the following example, the rewrite action replaces http in the Location header of an HTTP response with https if the response is associated with an HTTP redirect.
	`add rewrite action redloc replace 'http.res.header("Location").before_regex(re#://#)' '"https"'`
	`add rewrite policy pol1 HTTP.RES.IS_REDIRECT red_location`
	`bind rewrite global pol1 100`

Prefix	Description

SIP Expressions

Sep 23, 2014

The NetScaler default expressions language contains a number of expressions that operate on Session Initiation Protocol (SIP) connections. These expressions are intended to be used in policies for any supported protocol that operates on a request/response basis. (These expressions can be bound only to sip_udp virtual servers and global bind points.) These expressions can be used in content switching, rate limiting, responder, and rewrite policies.

Certain limitations apply to SIP expressions used with responder policies. The NetScaler operating system currently supports only SIP over UDP. Only the DROP, NOOP or RESPONDWITH actions are allowed on a SIP load balancing virtual server. Responder policies can be bound to a load balancing virtual server, an override global bind point, a default global bind point, or a sip_udp policy label.

The header format used by the SIP protocol is similar to that used by the HTTP protocol, so many of the new expressions look and function much like their HTTP analogs. Each SIP header consists of a line that includes the SIP method, the URL, and the version, followed by a series of name-value pairs that look like HTTP headers.

Following is a sample SIP header that is referred to in the expressions tables beneath it:

```
INVITE sip:16@www.sip.com:5060;transport=udp SIP/2.0
Record-Route: <sip:200.200.100.22;lr=on>
Via: SIP/2.0/UDP 200.200.100.22;branch=z9hG4bK444b.c8e103d1.0;rport=5060;
 received=10.102.84.18
Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;
 received=10.102.84.160
From: "12" <sip:12@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185
To: "16" <sip:16@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185
Call-ID: 00127f54-ec850017-0e46f5b9-5ec149c2@10.102.84.180
Max-Forwards: 69CSeq: 101 INVITE
User-Agent: Cisco-CP7940G/8.0
Contact: <sip:12@10.102.84.180:5060;transport=udp>
Expires: 180
Accept: application/sdp
Allow: ACK,BYE,CANCEL,INVITE,NOTIFY,OPTIONS,REFER,REGISTER,UPDATE
Supported: replaces,join,norefersub
Content-Length: 277
Content-Type: application/sdp
Content-Disposition: session;handling=optiona
```

The following tables contain lists of expressions that operate on SIP headers. The first table contains expressions that apply to request headers. Most response-based expressions are nearly the same as the corresponding request-based expressions. To create a response expression from the corresponding request expression, you change the first two sections of the expression from SIP.REQ to SIP.RES, and make other obvious adjustments. The second table contains those response expressions that are unique to responses and have no request equivalents. You can use any element in the following tables as a complete expression on its own, or you can use various operators to combine these expression elements with others to form more complex expressions.

Table 1. SIP Request Expressions

Expression	Description
SIP.REQ.METHOD	Operates on the method of the SIP request. The supported SIP request methods are ACK, BYE, CANCEL, INFO, INVITE, MESSAGE, NOTIFY, OPTIONS, PRACK, PUBLISH, REFER, REGISTER, SUBSCRIBE, and UPDATE. This expression is a derivative of the text class, so all operations that are applicable to text are applicable to this method. For example, for a SIP request of INVITE sip:16@10.102.84.181:5060;transport=udp SIP/2.0, this expression returns INVITE.
SIP.REQ.URL	Operates on the SIP request URL. This expression is a derivative of the text class, so all operations that are applicable to text are applicable to this method. For example, for a SIP request of INVITE sip:16@10.102.84.181:5060;transport=udp SIP/2.0, this expression returns sip:16@10.102.84.181:5060;transport=udp.
SIP.REQ.URL.PROTOCOL	Returns the URL protocol. For example, for a SIP URL of

Expression	Description
	sip:16@www.sip.com:5060;transport=udp, this expression returns sip.
SIP.REQ.URL.HOSTNAME	Returns the hostname portion of the SIP URL. For example, for a SIP URL of sip:16@www.sip.com:5060;transport=udp, this expression returns www.sip.com:5060.
SIP.REQ.URL.HOSTNAME.PORT	Returns the port portion of the SIP URL hostname. If no port is specified, this expression returns the default SIP port, 5060. For example, for a SIP hostname of www.sip.com:5060, this expression returns 5060.
SIP.REQ.URL.HOSTNAME.DOMAIN	Returns the domain name portion of the SIP URL hostname. If the host is an IP address, then this expression returns an incorrect result. For example, for a SIP hostname of www.sip.com:5060, this expression returns sip.com. For a SIP hostname of 192.168.43.15:5060, this expression returns an error.
SIP.REQ.URL.HOSTNAME.SERVER	Returns the server portion of the host. For example, for a SIP hostname of www.sip.com:5060, this expression returns www.
SIP.REQ.URL.USERNAME	Returns the username that precedes the @ character. For example, for a SIP URL of sip:16@www.sip.com:5060;transport=udp, this expression returns 16.
SIP.REQ.VERSION	Returns the SIP version number in the request. For example, for a SIP request of INVITE sip:16@10.102.84.181:5060;transport=udp SIP/2.0, this expression returns SIP/2.0.
SIP.REQ.VERSION.MAJOR	Returns the major version number (the number to the left of the period). For example, for a SIP version number of SIP/2.0, this expression returns 2.
SIP.REQ.VERSION.MINOR	Returns the minor version number (the number to the right of the period). For example, for a SIP version number of SIP/2.0, this expression returns 0.
SIP.REQ.CONTENT_LENGTH	Returns the contents of the Content-Length header. This expression is a derivative of the sip_header_t class, so all operations that are available for SIP headers can be used. For example, for a SIP Content-Length header of Content-Length: 277, this expression returns 277.
SIP.REQ.TO	Returns the contents of the To header. For example, for a SIP To header of To: "16" <sip:16@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns "16" <sip:16@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185.
SIP.REQ.TO.ADDRESS	Returns the SIP URI, which is found in the sip_url object. All operations that are available for SIP URIs can be used. For example, for a SIP To header of To: "16" <sip:16@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns sip:16@sip_example.com.
SIP.REQ.TO.DISPLAY_NAME	Returns the display name portion of the To header. For example, for a SIP To header of To: "16" <sip:16@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns 16.
SIP.REQ.TO.TAG	Returns the “tag” vale from the “tag” name value pair in the TO header. For example, for a SIP To header of To: "16" <sip:16@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns 00127f54ec85a6d90cc14f45-53cc0185.
SIP.REQ.FROM	Returns the contents of the From header. For example, for a SIP From header of From: "12" <sip:12@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns sip:12@sip_example.com.

Expression	Description
SIP.REQ.FROM.ADDRESS	Returns the SIP URI, which is found in the sip_url object. All operations that are available for SIP URIs can be used. For example, for a SIP From header of From: "12" <sip:12@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns sip:12@sip_example.com.
SIP.REQ.FROM.DISPLAY_NAME	Returns the display name portion of the To header. For example, for a SIP From header of From: "12" <sip:12@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns 12.
SIP.REQ.FROM.TAG	Returns the "tag" value from the "tag" name/value pair in the TO header. For example, for a SIP From header of From: "12" <sip:12@sip_example.com>;tag=00127f54ec85a6d90cc14f45-53cc0185, this expression returns 00127f54ec85a6d90cc14f45-53cc0185.
SIP.REQ.VIA	Returns the complete Via header. If there are multiple Via headers in the request, returns the last Via header. For example, for the two Via headers in the sample SIP header, this expression returns Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160.
SIP.REQ.VIA.SENTBY_ADDRESS	Returns the address that sent the request. For example, for the Via header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, this expression returns 10.102.84.180.
SIP.REQ.VIA.SENTBY_PORT	Returns the port that sent the request. For example, for the Via header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, this expression returns 5060.
SIP.REQ.VIA.RPORT	Returns the value from the rport name/value pair. For example, for the Via header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, this expression returns 5060.
SIP.REQ.VIA.BRANCH	Returns the value from the branch name/value pair. For example, for the Via header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, this expression returns z9hG4bK03e76d0b.
SIP.REQ.VIA.RECEIVED	Returns the value from the received name/value pair. For example, for the Via header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, this expression returns 10.102.84.160.
SIP.REQ.CALLID	Returns the contents of the Callid header. This expression is a derivative of the sip_header_t class, so all operations that are available for SIP headers can be used. For example, for a SIP Callid header of Call-ID: 00127f54-ec850017-0e46f5b9-5ec149c2@10.102.84.180, this expression returns 00127f54-ec850017-0e46f5b9-5ec149c2@10.102.84.180.
SIP.REQ.CSEQ	Returns the CSEQ number from the CSEQ, as an integer. For example, for a SIP CSEQ header of CSeq: 101 INVITE, this expression returns 101.
SIP.REQ.HEADER("<header_name>")	Returns the specified SIP header. For <header_name>, substitute the name of the header that you want. For example, to return the SIP From header, you would type SIP.REQ.HEADER("From").
SIP.REQ.HEADER("	Returns the specified instance of the specified SIP header. Multiple instances of the same SIP

Expression	Description
<header_name>").INSTANCE(<line_number>)	header can occur. Where you want a specific instance of such a SIP header (for example, a specific Via header), you can specify that header by typing a number as the <line_number>. Header instances are matched from last (0) to first. In other words, SIP.REQ.HEADER("Via").INSTANCE(0) returns the last instance of the Via header, while SIP.REQ.HEADER("Via").INSTANCE(1) returns the last instance but one of the Via header, and so on. For example, if used on the example SIP header, SIP.REQ.HEADER("Via").INSTANCE(1) returns Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060.
SIP.REQ.HEADER("<header_name>").VALUE(<line_number>)	Returns the contents of the specified instance of the specified SIP header. The usage is nearly the same as the previous expression. For example, if used on the SIP header example in the preceding table entry, SIP.REQ.HEADER("Via").VALUE(1) returns SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060.
SIP.REQ.HEADER("<header_name>").COUNT	Returns the number of instances of a particular header as an integer. For example, if used on the SIP header example above, SIP.REQ.HEADER("Via").COUNT returns 2.
SIP.REQ.HEADER("<header_name>").EXISTS	Returns a boolean value of true or false, depending upon whether the specified header exists or not. For example, if used on the SIP header example above, SIP.REQ.HEADER("Expires").EXISTS returns true, while SIP.REQ.HEADER("Caller-ID").EXISTS returns false.
SIP.REQ.HEADER("<header_name>").LIST	Returns the comma-separated parameter list in the specified header. For example, if used on the SIP header example above, SIP.REQ.HEADER("Allow").LIST returns ACK,BYE,CANCEL,INVITE,NOTIFY,OPTIONS,REFER,REGISTER,UPDATE. You can append the string .GET(<list_item_number>) to select a specific list item. For example, to get the first item (ACK) from the above list, you would type SIP.REQ.HEADER("Allow").LIST.GET(0). To extract the second item (BYE), you would type SIP.REQ.HEADER("Allow").LIST.GET(1). Note: If the specified header contains a list of name/value pairs, the entire name/value pair is returned.
SIP.REQ.HEADER("<header_name>").TYPECAST_SIP_HEADER_T("<in_header_name>")	Typecasts <header_name> to <in_header_name>. Any text can be typecasted to the sip_header_t class, after which all header-based operations can be used. After you perform this operation, you can apply all operations that can be used with <in_header_name>. For example, the expression SIP.REQ.CONTENT_LENGTH.TYPECAST_SIP_HEADER_T typecasts all instances of the Content-Length header. After you perform this operation, you can apply all header operations to all instances of the specified header.
SIP.REQ.HEADER("<header_name>").CONTAINS("<string>").	Returns boolean true if the specified text string is present in any instance of the specified header. Operates on all the instances of the specified header. Header instances are matched from last (0) to first.
SIP.REQ.HEADER("<header_name>").EQUALS_ANY(<patset>)	Returns boolean true if any pattern associated with <patset> matches any content in any instance of the specified header. Operates on all the instances of the specified header. Header instances are matched from last (0) to first.
SIP.REQ.HEADER("<header_name>").CONTAINS_ANY(<patset>)	Returns Boolean true if any pattern associated with <patset> matches any content in any instance of the specified header. Operates on all the instances of the specified header. Header instances are matched from last (0) to first.
SIP.REQ.HEADER("<header_name>").CONTAINS_INDEX(<patset>)	Returns the index of the matching pattern associated with <patset> if that pattern matches any content in any instance of the specified header. Operates on all the instances of the specified header. Header instances are matched from last (0) to first.

Expression	Description
SIP.REQ.HEADER("<header_name>").EQUALS_INDEX(<patset>)	Returns the index of the matching pattern associated with <patset> if that pattern matches any instance of the specified header. Operates on all the instances of the specified header. Header instances are matched from last (0) to first.
SIP.REQ.HEADER("<header_name>").SUBSTR("<string>")	If the specified string is present in any instance of the specified header, this expression returns that string. For example, for the SIP header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160", SIP.REQ.HEADER("Via").SUBSTR("rport=5060") returns "rport=5060". SIP.REQ.HEADER("Via").SUBSTR("rport=5061") returns an empty string.
SIP.REQ.HEADER("<header_name>").AFTER_STR("<string>")	If the specified string is present in any instance of the specified header, this expression returns the string immediately after that string. For example, for the SIP header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, the expression SIP.REQ.HEADER("Via").AFTER_STR("rport=") returns 5060.
SIP.REQ.HEADER("<header_name>").REGEX_MATCH(<regex>)	Returns boolean true if the specified regular expression (*regex*) matches any instance of the specified header. You must specify the regular expression in the following format: re<delimiter>regular expression<same delimiter> The regular expression cannot be larger than 1499 characters in length. It must conform to the PCRE regular expression library. See http://www.pcre.org/pcre.txt for documentation on PCRE regular expression syntax. The pcrepattern man page also has useful information on specifying patterns by using PCRE regular expressions. The regular expression syntax supported in this expression has some differences from PCRE. Back references are not allowed. You should avoid recursive regular expressions; although some work, many do not. The dot (.) metacharacter matches newlines. Unicode is not supported. SET_TEXT_MODE(IGNORECASE) overrides the (?i) internal option specified in the regular expression.
SIP.REQ.HEADER("<header_name>").REGEX_SELECT(<regex>)	If the specified regex matches any text in any instance of the specified header, this expression returns the text. For example, for the SIP header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, the expression SIP.REQ.HEADER("Via").REGEX_SELECT("received=[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}") returns received=10.102.84.160.
SIP.REQ.HEADER("<header_name>").AFTER_REGEX(<regex>)	If the specified regex matches any text in any instance of the specified header, this expression returns the string immediately after that text. For example, for the SIP header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, the expression SIP.REQ.HEADER("Via").AFTER_REGEX("received=") returns 10.102.84.160.
SIP.REQ.HEADER("<header_name>").BEFORE_REGEX(<regex>)	If the specified regex matches any text in any instance of the specified header, this expression returns the string immediately before that text. For example, for the SIP header Via: SIP/2.0/UDP 10.102.84.180:5060;branch=z9hG4bK03e76d0b;rport=5060;received=10.102.84.160, the expression SIP.REQ.HEADER("Via").BEFORE_REGEX("[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}") returns received=.
SIP.REQ.FULL_HEADER	Returns the entire SIP header, including the terminating CR/LF.
SIP.REQ.IS_VALID	Returns boolean true if the request format is valid.
SIP.REQ.BODY(<length>)	Returns the request body, up to the specified length. If the specified length is greater than the length of the request body, this expression returns the entire request body.
SIP.REQ.LB_VSERVER	Returns the name of the load balancing virtual server (*LB vserver*) that is serving the current request.

Expression	Description
SIP.REQ.CS_VSERVER	Returns the name of the content switching virtual server (*CS vserver*) that is serving the current request.

Table 2. SIP Response Expressions

Expression	Description
SIP.RES.STATUS	Returns the SIP response status code. For example, if the first line of the response is SIP/2.0 100 Trying, this expression returns 100.
SIP.RES.STATUS_MSG	Returns the SIP response status message. For example, if the first line of the response is SIP/2.0 100 Trying, this expression returns Trying.
SIP.RES.IS_REDIRECT	Returns boolean true if the response code is a redirect.
SIP.RES.METHOD	Returns the response method extracted from the request method string in the CSeq header.

Operations for HTTP, HTML, and XML Encoding and “Safe” Characters

Nov 27, 2014

The following operations work with the encoding of HTML data in a request or response and XML data in a POST body.

Table 1. Operations That Evaluate HTML and XML Encoding

HTML or XML Operation	Description
<text>.HTML_XML_SAFE	Transforms special characters into XML safe format, as in the following examples: • A left-pointing angle bracket (<) is converted to < • A right-pointing angle bracket (>) is converted to > • An ampersand (&) is converted to & This operation safeguards against cross-site scripting attacks. Maximum length of the transformed text is 2048 bytes. This is a read-only operation. After applying the transformation, additional operators that you specify in the expression are applied to the selected text. Following is an example: `http.req.url.query.html_xml_safe.contains("myQueryString")`
<text>.HTTP_HEADER_SAFE	Converts all new line ('\n') characters in the input text to '%0A' to enable the input to be used safely in HTTP headers. This operation safeguards against response-splitting attacks. The maximum length of the transformed text is 2048 bytes. This is a read-only operation.
<text>.HTTP_URL_SAFE	Converts unsafe URL characters to '%xx' values, where “xx” is a hex-based representation of the input character. For example, the ampersand (&) is represented as %26 in URL-safe encoding. The maximum length of the transformed text is 2048 bytes. This is a read-only operation. Following are URL safe characters. All others are unsafe: • Alpha-numeric characters: a-z, A-Z, 0-9 • Asterix: "*" • Ampersand: "&" • At-sign: "@" • Colon: ":" • Comma: "," • Dollar: "$" • Dot: "." • Equals: "=" • Exclamation mark: "!" • Hyphen: "-" • Open and close parentheses: "(", ")" • Percent: "%" • Plus: "+" • Semicolon: ";" • Single quote: "'" • Slash: "/" • Question mark: "?" • Tilde: "~" • Underscore: "_"
<text>.MARK_SAFE	Marks the text as safe without applying any type of data transformation.

HTML or XML Operation	Description
<text>.SET_TEXT_MODE(URLENCODED|NOURLENCODED)	Transforms all %HH encoding in the byte stream. This operation works with characters (not bytes). By default, a single byte represents a character in ASCII encoding. However, if you specify URLENCODED mode, three bytes can represent a character. In the following example, a PREFIX(3) operation selects the first 3 characters in a target. http.req.url.hostname.prefix(3) In the following example, the NetScaler can select up to 9 bytes from the target: http.req.url.hostname.set_text_mode(urlencoded).prefix(3)
<text>.SET_TEXT_MODE(PLUS_AS_SPACE|NO_PLUS_AS_SPACE)	Specifies how to treat the plus character (+). The PLUS_AS_SPACE option replaces a plus character with white space. For example, the text “hello+world” becomes “hello world.” The NO_PLUS_AS_SPACE option leaves plus characters as they are.
<text>.SET_TEXT_MODE(BACKSLASH_ENCODED|NO_BACKSLASH_ENCODED)	Specifies whether or not backslash decoding is performed on the text object represented by <text>. If BACKSLASH_ENCODED is specified, the SET_TEXT_MODE operator performs the following operations on the text object: • All occurrences of “\XXX” will be replaced with the character “Y” (where XXX represents a number in the octal system and Y represents the ASCII equivalent of XXX). The valid range of octal values for this type of encoding is 0 to 377. For example, the encoded text "http\72//" and "http\072//" will both be decoded to "http://", where the colon (:) is the ASCII equivalent of the octal value “72”. • All occurrences of “\xHH” will be replaced with the character “Y” (HH represents a number in the hexadecimal system and Y denotes the ASCII equivalent of HH. For example, the encoded text "http\x3a//" will be decoded to "http://", where the colon (:) is the ASCII equivalent of the hexadecimal value “3a“. • All occurrences of “\\uWWXX” will be replaced with the character sequence “YZ” (Where WW and XX represent two distinct hexadecimal values and Y and Z represent their ASCII equivalents of WW and XX respectively. For example, the encoded text "http%u3a2f/" and "http%u003a//" will both be decoded to "http://", where “3a” and “2f” are two hexadecimal values and the colon (:) and forward slash (“/”) represent their ASCII equivalents respectively. • All occurrences of "\b", "\n", "\t", "\f", and "\r" are replaced with the corresponding ASCII characters. If NO_BACKSLASH_ENCODED is specified, backslash decoding is not performed on the text object.
<text>.SET_TEXT_MODE(BAD_ENCODE_RAISE_UNDEF|NO_BAD_ENCODE_RAISE_UNDEF)	Performs the associated undefined action if either the URLENCODED or the BACKSLASH_ENCODED mode is set and bad encoding corresponding to the specified encoding mode is encountered in the text object represented by <text>. If NO_BAD_ENCODE_RAISE_UNDEF is specified, the associated undefined action will not be performed when bad encoding is encountered in the text object represented by<text>.

Expressions for TCP, UDP, and VLAN Data

Sep 02, 2013

TCP and UDP data take the form of a string or a number. For expression prefixes that return string values for TCP and UDP data, you can apply any text-based operations. For more information, see "Default Syntax Expressions: Evaluating Text."

For expression prefixes that return numeric value, such as a source port, you can apply an arithmetic operation. For more information, see "Basic Operations on Expression Prefixes" and "Compound Operations for Numbers."

The following table describes prefixes that extract TCP and UDP data.

Table 1. Prefixes That Extract TCP and UDP Data

GET Operation	Description
CLIENT.TCP.PAYLOAD(<integer>)	Returns TCP payload data as a string, starting with the first character in the payload and continuing for the number of characters in the <integer> argument. You can apply any text-based operation to this prefix.
CLIENT.TCP.SRCPORT	Returns the ID of the current packet's source port as a number.
CLIENT.TCP.DSTPORT	Returns the ID of the current packet's destination port as a number.
CLIENT.TCP.OPTIONS	Returns the TCP options set by the client. Examples of TCP options are Maximum Segment Size (MSS), Window Scale, Selective Acknowledgements (SACK), and Time Stamp Option. The COUNT, TYPE(<type>), and TYPE_NAME(<m>) operators can be used with this prefix. For the TCP options set by the server, see the SERVER.TCP.OPTIONS prefix.
CLIENT.TCP.OPTIONS.COUNT	Returns the number of TCP options that the client has set.
CLIENT.TCP.OPTIONS.TYPE(<type>)	Returns the value of the TCP option whose type (or *option kind*) is specified as the argument. The value is returned as a string of bytes in big endian format (or *network byte order*). **Parameters:** type - Type value
CLIENT.TCP.OPTIONS.TYPE_NAME(<m>)	Returns the value of the TCP option whose enumeration constant is specified as the argument. The enumeration constants that you can

GET Operation	Description
	pass as the argument are REPEATER, TIMESTAMP, SACK_PERMITTED, WINDOW, and MAXSEG. To specify the TCP option kind instead of these enumeration constants, use CLIENT.TCP.OPTIONS.TYPE(<type>). For other TCP options, you must use CLIENT.TCP.OPTIONS.TYPE(<type>). Parameters: m - TCP option enumeration constant
CLIENT.TCP.REPEATER_OPTION.EXISTS	Returns a Boolean TRUE if Repeater TCP options exist.
CLIENT.TCP.REPEATER_OPTION.IP	Returns the branch repeater's IPv4 address from the Repeater TCP options.
CLIENT.TCP.REPEATER_OPTION.MAC	Returns the branch repeater's MAC address from the Repeater TCP options.
CLIENT.UDP.DNS.DOMAIN	Returns the DNS domain name.
CLIENT.UDP.DNS.DOMAIN.EQ("<hostname>")	Returns a Boolean TRUE if the domain name matches the <hostname> argument. The comparison is case insensitive. Following is an example: client.udp.dns.domain.eq("www.mycompany.com")
CLIENT.UDP.DNS.IS_AAAAREC	Returns a Boolean TRUE if the record type is AAAA. These types of records indicate an IPv6 address in forward lookups.
CLIENT.UDP.DNS.IS_ANYREC	Returns a Boolean TRUE if it is of any record type.
CLIENT.UDP.DNS.IS_AREC	Returns a Boolean TRUE if the record is type A. Type A records provide the host address.
CLIENT.UDP.DNS.IS_CNAMEREC	Returns a Boolean TRUE if the record is of type CNAME. In systems that use multiple names to identify a resource, there is one canonical name and a number of aliases. The CNAME provides the canonical name.
CLIENT.UDP.DNS.IS_MXREC	Returns a Boolean TRUE if the record is of type MX (mail exchanger). This DNS record describes a priority and a host name. The MX records for the same domain name specify the email servers in the

GET Operation	Description
	domain and the priority for each server.
CLIENT.UDP.DNS.IS_NSREC	Returns a Boolean TRUE if the record is of type NS. This is a name server record that includes a host name with an associated A record. This enables locating the domain name that is associated with the NS record.
CLIENT.UDP.DNS.IS_PTRREC	Returns a Boolean TRUE if the record is of type PTR. This is a domain name pointer and is often used to associate a domain name with an IPv4 address.
CLIENT.UDP.DNS.IS_SOAREC	Returns a Boolean TRUE if the record is of type SOA. This is a start of authority record.
CLIENT.UDP.DNS.IS_SRVREC	Returns a Boolean TRUE if the record is of type SRV. This is a more general version of the MX record.
CLIENT.UDP.DSTPORT	Returns the numeric ID of the current packet's UDP destination port.
CLIENT.UDP.SRCPORT	Returns the numeric ID of the current packet's UDP source port.
CLIENT.UDP.RADIUS	Returns RADIUS data for the current packet.
CLIENT.UDP.RADIUS.ATTR_TYPE(<type>)	Returns the value for the attribute type specified as the argument.
CLIENT.UDP.RADIUS.USERNAME	Returns the RADIUS user name.
CLIENT.TCP.MSS	Returns the maximum segment size (MSS) for the current connection as a number.
CLIENT.VLAN.ID	Returns the numeric ID of the VLAN through which the current packet entered the NetScaler.
SERVER.TCP.DSTPORT	Returns the numeric ID of the current packet's destination port.
SERVER.TCP.SRCPORT	Returns the numeric ID of the current packet's source port.
SERVER.TCP.OPTIONS	Returns the TCP options set by the server. Examples of TCP options are Maximum Segment Size (MSS), Window Scale, Selective Acknowledgements (SACK), and Time Stamp Option. The COUNT, TYPE(<type>), and TYPE_NAME(<m>) operators can be used

GET Operation	Description
	with this prefix. For the TCP options set by the client, see the CLIENT.TCP.OPTIONS prefix.
SERVER.TCP.OPTIONS.COUNT	Returns the number of TCP options that the server has set.
SERVER.TCP.OPTIONS.TYPE(<type>)	Returns the value of the TCP option whose type (or *option kind*) is specified as the argument. The value is returned as a string of bytes in big endian format (or *network byte order*). **Parameters:** type - Type value
SERVER.TCP.OPTIONS.TYPE_NAME(<m>)	Returns the value of the TCP option whose enumeration constant is specified as the argument. The enumeration constants that you can pass as the argument are REPEATER, TIMESTAMP, SACK_PERMITTED, WINDOW, and MAXSEG. To specify the TCP option kind instead of these enumeration constants, use CLIENT.TCP.OPTIONS.TYPE(<type>). For other TCP options, you must use CLIENT.TCP.OPTIONS.TYPE(<type>). **Parameters:** m - TCP option enumeration constant
SERVER.VLAN	Operates on the VLAN through which the current packet entered the NetScaler.
SERVER.VLAN.ID	Returns the numeric ID of the VLAN through which the current packet entered the NetScaler.

Expressions for Evaluating a DNS Message and Identifying Its Carrier Protocol

Aug 30, 2013

You can evaluate DNS requests and responses by using expressions that begin with DNS.REQ and DNS.RES, respectively. You can also identify the transport layer protocol that is being used to send the DNS messages.

The following functions return the contents of a DNS query.

Table 1. Functions that return the contents of a DNS query

Function	Description
DNS.REQ.QUESTION.DOMAIN	Return the domain name (the value of the QNAME field) in the question section of the DNS query. The domain name is returned as a text string, which can be passed to EQ(), NE(), and any other functions that work with text.
DNS.REQ.QUESTION.TYPE	Return the query type (the value of the QTYPE field) in the DNS query. The field indicates the type of resource record (for example, A, NS, or CNAME) for which the name server is being queried. The returned value can be compared to one of the following values by using the EQ() and NE() functions: • A • AAAA • NS • SRV • PTR • CNAME • SOA • MX • ANY Note: You can use only the EQ() and NE() functions with the TYPE function. **Example:** DNS.REQ.QUESTION.TYPE.EQ(MX)

The following functions return the contents of a DNS response.

Table 2. Functions that return the contents of a DNS response

Function	Description
DNS.RES.HEADER.RCODE	Return the response code (the value of the RCODE field) in the header section of the DNS response. You can use only the EQ() and NE() functions with the

Function	Description
	RCODE function. Following are the possible values: • NOERROR • FORMERR • SERVFAIL • NXDOMAIN • NOTIMP • REFUSED
DNS.RES.QUESTION.DOMAIN	Return the domain name (the value of the QNAME field) in the question section of the DNS response. The domain name is returned as a text string, which can be passed to EQ(), NE(), and any other functions that work with text.
DNS.RES.QUESTION.TYPE	Return the query type (the value of the QTYPE field) in the question section of the DNS response. The field indicates the type of resource record (for example, A, NS, or CNAME) that is contained in the response. The returned value can be compared to one of the following values by using the EQ() and NE() functions: • A • AAAA • NS • SRV • PTR • CNAME • SOA • MX • ANY You can use only the EQ() and NE() functions with the TYPE function. Example: DNS.RES.QUESTION.TYPE.EQ(SOA)

The following functions return the transport layer protocol name.

Table 3. Functions that return the transport layer protocol name

Function	Description
DNS.REQ.TRANSPORT	Return the name of the transport layer protocol that was used to send the DNS query. Possible values returned are TCP and UDP. You can use only the EQ() and NE() functions with the TRANSPORT function. Example: DNS.REQ.TRANSPORT.EQ(TCP)
DNS.RES.TRANSPORT	Return the name of the transport layer protocol that was used for the DNS response.

Function	Description
	Possible values returned are TCP and UDP. You can use only the EQ() and NE() functions with the TRANSPORT function. **Example:** DNS.RES.TRANSPORT.EQ(TCP)

XPath and HTML, XML, or JSON Expressions

Mar 20, 2012

The default syntax expression engine supports expressions for evaluating and retrieving data from HTML, XML, and JavaScript Object Notation (JSON) files. This enables you to find specific nodes in an HTML, XML, or JSON document, determine if a node exists in the file, locate nodes in XML contexts (for example, nodes that have specific parents or a specific attribute with a given value), and return the contents of such nodes. Additionally, you can use XPath expressions in rewrite expressions.

The default syntax expression implementation for XPath comprises a default syntax expression prefix (such as "HTTP.REQ.BODY") that designates HTML or XML text, and the XPATH operator that takes the XPath expression as its argument.

HTML files are a largely free-form collection of tags and text elements. You can use the XPATH_HTML operator, which takes an XPath expression as its argument, to process HTML files. JSON files are either a collection of name/value pairs or an ordered list of values. You can use the XPATH_JSON operator, which takes an XPath expression as its argument, to process JSON files.

Table 1. XPath and JSON Expression Prefixes That Return Text

XPath Prefix	Description
<text>.XPATH(xpathex)	Operate on an XML file and return a Boolean value. For example, the following expression returns a Boolean TRUE if a node called "creator" exists under the node "Book" within the first 1000 bytes of the XML file. HTTP.REQ.BODY(1000).XPATH(xp%boolean(//Book/creator)%) Parameters: xpathex - XPath Boolean expression
<text>.XPATH(xpathex)	Operate on an XML file and return a value of data type "double." For example, the following expression converts the string "36" (a price value) to a value of data type "double" if the string is in the first 1000 bytes of the XML file: HTTP.REQ.BODY(1000).XPATH(xp%number(/Book/price)%) Parameters: xpathex - XPath numeric expression
<text>.XPATH(xpathex)	Operate on an XML file and return a node-set or a string. Node-sets are converted to corresponding strings by using the standard XPath string conversion routine. For example, the following expression selects all the nodes that are enclosed by "/Book/creator" (a node-set) in the first 1000 bytes of the body: HTTP.REQ.BODY(1000).XPATH(xp%/Book/creator%) Parameters: xpathex - XPath expression
<text>.XPATH_HTML(xpathex)	Operate on an HTML file and return a text value. For example, the following expression operates on an HTML file and returns the

XPath Prefix	Description
	text enclosed in <title></title> tags if the title HTML element is found in the first 1000 bytes: HTTP.REQ.BODY(1000).XPATH_HTML(xp%/html/head/title%) Parameters: xpathex - XPath text expression
<text>.XPATH_HTML_WITH_MARKUP(xpathex)	Operate on an HTML file and return a string that contains the entire selected portion of the document, including markup such as including the enclosing element tags. The following expression operates on the HTML file and selects all content within the <title> tag, including markup. HTTP.REQ.BODY(1000).XPATH_HTML_WITH_MARKUP(xp%/html/head/title%) The portion of the HTML body that is selected by the expression is marked for further processing. Parameters: xpathex - XPath expression
<text>.XPATH_JSON(xpathex)	Operate on a JSON file and return a Boolean value. For example, consider the following JSON file: { "Book":{ "creator":{ "person":{ "name":'<name>' } }, "title":'<title>' } } The following expression operates on the JSON file and returns a Boolean TRUE if the JSON file contains a node named “creator,” whose parent node is “Book,” in the first 1000 bytes: HTTP.REQ.BODY(1000).XPATH_JSON(xp%boolean(/Book/creator)%) Parameters: xpathex - XPath Boolean expression
<text>.XPATH_JSON(xpathex)	Operate on a JSON file and return a value of data type “double.” For example, consider the following JSON file: { "Book":{ "creator":{ "person":{ "name":'<name>' } }, "title":'<title>', "price":"36" } } The following expression operates on the JSON file and converts the string “36” to a value of data type “double” if the string is present in the first 1000 bytes of the JSON file. HTTP.REQ.BODY(1000).XPATH_JSON(xp%number(/Book/price)%) Parameters: xpathex - XPath numeric expression
<text>.XPATH_JSON(xpathex)	Operate on a JSON file and return a node-set or a string. Node-sets are converted to corresponding strings by using the standard XPath string conversion routine.

XPath Prefix	Description
	For example, consider the following JSON file: { "Book":{ "creator":{ "person":{ "name":'<name>' } }, "title":'<title>' } } The following expression selects all the nodes that are enclosed by “/Book" (a node-set) in the first 1000 bytes of the body of the JSON file and returns the corresponding string value, which is "<name><title>": HTTP.REQ.BODY(1000).XPATH_JSON(xp%/Book%) Parameters: xpathex - XPath expression
<text>.XPATH_JSON_WITH_MARKUP(xpathex)	Operate on an XML file and return a string that contains the entire portion of the document for the result node, including markup such as including the enclosing element tags. For example, consider the following JSON file: {"Book":{ "creator":{ "person":{ "name":'<name>' } }, "title":'<title>' } } The following expression operates on the JSON file and selects all the nodes that are enclosed by “/Book/creator" in the first 1000 bytes of the body, which is “creator:{ person:{ name:'<name>' } }.” HTTP.REQ.BODY(1000).XPATH_JSON_WITH_MARKUP(xp%/Book/creator%) The portion of the JSON body that is selected by the expression is marked for further processing. Parameters: xpathex - XPath expression
<text>.XPATH_WITH_MARKUP(xpathex)	Operate on an XML file and return a string that contains the entire portion of the document for the result node, including markup such as including the enclosing element tags. For example, the following expression operates on an XML file and selects all the nodes enclosed by “/Book/creator" in the first 1000 bytes of the body. HTTP.REQ.BODY(1000).XPATH_WITH_MARKUP(xp%/Book/creator%) The portion of the JSON body that is selected by the expression is marked for further processing. Parameters: xpathex - XPath expression

Encrypting and Decrypting XML Payloads

Sep 02, 2013

You can use the XML_ENCRYPT() and XML_DECRYPT() functions in default syntax expressions to encrypt and decrypt, respectively, XML data. These functions conform to the W3C XML Encryption standard defined at "http://www.w3.org/TR/2001/PR-xmldsig-core-20010820/." XML_ENCRYPT() and XML_DECRYPT() support a subset of the XML Encryption specification. In the subset, data encryption uses a bulk cipher method (RC4, DES3, AES128, AES192, or AES256), and an RSA public key is used to encrypt the bulk cipher key.

Note: If you want to encrypt and decrypt text in a payload, you must use the ENCRYPT and DECRYPT functions. For more information about these functions, see "Encrypting and Decrypting Text."

The XML_ENCRYPT() and XML_DECRYPT() functions are not dependent on the encryption/decryption service that is used by the ENCRYPT and DECRYPT commands for text. The cipher method is specified explicitly as an argument to the XML_ENCRYPT() function. The XML_DECRYPT() function obtains the information about the specified cipher method from the <xenc:EncryptedData> element. Following are synopses of the XML encryption and decryption functions:

- **XML_ENCRYPT(<certKeyName>, <method> [, <flags>])**. Returns an <xenc:EncryptedData> element that contains the encrypted input text and the encryption key, which is itself encrypted by using RSA.
- **XML_DECRYPT(<certKeyName>)**. Returns the decrypted text from the input <xenc:EncryptedData> element, which includes the cipher method and the RSA-encrypted key.

Note: The <xenc:EncryptedData> element is defined in the W3C XML Encryption specification.

Following are descriptions of the arguments:

certKeyName

Selects an X.509 certificate with an RSA public key for XML_ENCRYPT() or an RSA private key for XML_DECRYPT(). The certificate key must have been previously created by an add ssl certKey command.

method

Specifies which cipher method to use for encrypting the XML data. Possible values: RC4, DES3, AES128, AES192, AES256.

flags

A bitmask specifying the following optional key information (<ds:KeyInfo>) to be included in the <xenc:EncryptedData> element that is generated by XML_ENCRYPT():

- **1** - Include a KeyName element with the certKeyName. The element is <ds:KeyName>.
- **2** - Include a KeyValue element with the RSA public key from the certificate. The element is <ds:KeyValue>.
- **4** - Include an X509IssuerSerial element with the certificate serial number and issuer DN. The element is <ds:X509IssuserSerial>.
- **8** - Include an X509SubjectName element with the certificate subject DN. The element is <ds:X509SubjectName>.
- **16** - Include an X509Certificate element with the entire certificate. The element is <ds:X509Certificate>.

XML_ENCRYPT() XML_DECRYPT()

The XML encryption feature uses SSL certificate-key pairs to provide X.509 certificates (with RSA public keys) for key encryption and RSA private keys for key decryption. Therefore, before you use the XML_ENCRYPT() function in an expression, you must create an SSL certificate-key pair. The following command creates an SSL certificate-key pair, my-

certkey, with the X.509 certificate, my-cert.pem, and the private key file, my-key.pem.

```
add ssl certKey my-certkey -cert my-cert.pem -key my-key.pem -passcrypt kxPeMRYnitY=
```

The following CLI commands create rewrite actions and policies for encrypting and decrypting XML content.

```
add rewrite action my-xml-encrypt-action replace "HTTP.RES.BODY(10000).XPATH_WITH_MARKUP(xp%/%)"
"HTTP.RES.BODY(10000).XPATH_WITH_MARKUP(xp%/%).XML_ENCRYPT(\"my-certkey\", AES256, 31)" -
bypassSafetyCheck YES

add rewrite action my-xml-decrypt-action replace
"HTTP.REQ.BODY(10000).XPATH_WITH_MARKUP(xp%//xenc:EncryptedData%)"
"HTTP.REQ.BODY(10000).XPATH_WITH_MARKUP(xp%//xenc:EncryptedData%).XML_DECRYPT(\"my-
certkey\")" -bypassSafetyCheck YES

add rewrite policy my-xml-encrypt-policy "HTTP.REQ.URL.CONTAINS(\"xml-encrypt\")" my-xml-encrypt-
action

add rewrite policy my-xml-decrypt-policy
"HTTP.REQ.BODY(10000).XPATH(xp%boolean(//xenc:EncryptedData)%)" my-xml-decrypt-action

bind rewrite global my-xml-encrypt-policy 30

bind rewrite global my-xml-decrypt-policy 30
```

In the above example, the rewrite action my-xml-encrypt-action encrypts the entire XML document (XPATH_WITH_MARKUP(xp%/%)) in the request by using the AES-256 bulk encryption method and the RSA public key from my-certkey to encrypt the bulk encryption key. The action replaces the document with an <xenc:EncryptedData> element containing the encrypted data and an encrypted key. The flags represented by 31 include all of the optional <ds:KeyInfo> elements.

The action my-xml-decrypt-action decrypts the first <xenc:EncryptedData> element in the response (XPATH_WITH_MARKUP(xp%//xenc:EncryptedData%)). This requires the prior addition of the xenc XML namespace by use of the following CLI command:

```
add ns xmlnamespace xenc http://www.w3.org/2001/04/xmlenc#
```

The my-xml-decrypt-action action uses the RSA private key in my-certkey to decrypt the encrypted key and then uses the bulk encryption method specified in the element to decrypt the encrypted contents. Finally, the action replaces the encrypted data element with the decrypted content.

The rewrite policy my-xml-encrypt-policy applies my-xml-encrypt-action to requests for URLs containing xml-encrypt. The action encrypts the entire response from a service configured on the NetScaler appliance.

The rewrite policy my-xml-decrypt-policy applies my-xml-decrypt-action to requests that contain an <xenc:EncryptedData> element ((XPATH(xp%//xenc:EncryptedData%) returns a non-empty string). The action decrypts the encrypted data in requests that are bound for a service configured on the NetScaler appliance.

Default Syntax Expressions: Parsing SSL Certificates

May 25, 2015

You can use default syntax expressions to evaluate X.509 Secure Sockets Layer (SSL) client certificates. A client certificate is an electronic document that can be used to authenticate a user's identity. A client certificate contains (at a minimum) version information, a serial number, a signature algorithm ID, an issuer name, a validity period, a subject (user) name, a public key, and signatures.

You can examine both SSL connections and data in client certificates. For example, you may want to send SSL requests that use low-strength ciphers to a particular load balancing virtual server farm. The following command is an example of a Content Switching policy that parses the cipher strength in a request and matches cipher strengths that are less than or equal to 40:

```
add cs policy p1 -rule "client.ssl.cipher_bits.le(40)"
```

As another example, you can configure a policy that determines whether a request contains a client certificate:

```
add cs policy p2 -rule "client.ssl.client_cert EXISTS"
```

Or, you might want to configure a policy that examines particular information in a client certificate. For example, the following policy verifies that the certificate has one or more days before expiration:

```
add cs policy p2 -rule "client.ssl.client_cert exists && client.ssl.client_cert.days_to_expire.ge(1)"
```

Note: For information on parsing dates and times in a certificate, see "Format of Dates and Times in an Expression" and "Expressions for SSL Certificate Dates."
This document includes the following details:

- Prefixes for Text-Based SSL and Certificate Data
- Prefixes for Numeric Data in SSL Certificates
- Expressions for SSL Certificates

The following table describes expression prefixes that identify text-based items in SSL transactions and client certificates.

Table 1. Prefixes That Return Text or Boolean Values for SSL and Client Certificate Data

Prefix	Description
CLIENT.SSL.CLIENT_CERT	Returns the SSL client certificate in the current SSL transaction.
CLIENT.SSL.CLIENT_CERT.TO_PEM	Returns the SSL client certificate in binary format.
CLIENT.SSL.CIPHER_EXPORTABLE	Returns a Boolean TRUE if the SSL cryptographic SSL cryptographic cipher is exportable.
CLIENT.SSL.CIPHER_NAME	Returns the name of the SSL Cipher if invoked from an SSL connection, and a NULL string if invoked from a non-SSL connection.
CLIENT.SSL.IS_SSL	Returns a Boolean TRUE if the current connection is SSL-based.

Updated: 2015-06-17

The following table describes prefixes that evaluate numeric data other than dates in SSL certificates. These prefixes can be used with the operations that are described in "Basic Operations on Expression Prefixes" and "Compound Operations for Numbers."

Table 2. Prefixes That Evaluate Numeric Data Other Than Dates in SSL Certificates

Prefix	Description
CLIENT.SSL.CLIENT_CERT.DAYS_TO_EXPIRE	Returns the number of days that the certificate is valid, or returns -1 for expired certificates.
CLIENT.SSL.CLIENT_CERT.PK_SIZE	Returns the size of the public key used in the certificate.
CLIENT.SSL.CLIENT_CERT.VERSION	Returns the version number of the certificate. If the connection is not SSL-based, returns zero (0).
CLIENT.SSL.CIPHER_BITS	Returns the number of bits in the cryptograhic key. Returns 0 if the connection is not SSL-based.
CLIENT.SSL.VERSION	Returns a number that represents the SSL protocol version, as follows: • 0. The transaction is not SSL-based. • 0x002. The transaction is SSLv2. • 0x300. The transaction is SSLv3. • 0x301. The transaction is TLSv1. • 0x302. The transaction is TLSv1.1. • 0x303. The transaction is TLSv1.2.

Note: For expressions related to expiration dates in a certificate, see "Expressions for SSL Certificate Dates."

Updated: 2013-09-02

You can parse SSL certificates by configuring expressions that use the following prefix:

CLIENT.SSL.CLIENT_CERT

This section discusses the expressions that you can configure for certificates, with the exception of expressions that examine certificate expiration. Time-based operations are described in "Default Syntax Expressions: Working with Dates, Times, and Numbers."

The following table describes operations that you can specify for the CLIENT.SSL.CLIENT_CERT prefix.

Table 3. Operations That Can Be Specified with the CLIENT.SSL.CLIENT_CERT Prefix

SSL Certificate Operation	Description
<certificate>.EXISTS	Returns a Boolean TRUE if the client has an SSL certificate.
<certificate>.ISSUER	Returns the Distinguished Name (DN) of the Issuer in the certificate as a name-value list. An equals sign (“=”) is th name and the value, and the slash (“/”) is the delimiter that separates the name-value pairs. Following is an example of the returned DN: /C=US/O=myCompany/OU=www.mycompany.com/CN=www.mycompany.com/emailAddress=myuser
<certificate>.ISSUER. IGNORE_EMPTY_ELEMENTS	Returns the Issuer and ignores the empty elements in a name-value list. For example, consider the following: Cert-Issuer: /c=in/st=kar//l=bangelore //o=mycompany/ou=sales/ /emailAddress=myuserid@mycc The following Rewrite action returns a count of 6 based on the preceding Issuer definition: sh rewrite action insert_ssl_header Name: insert_ssl Operation: insert_http_header Target:Cert-Issuer Value:CLIENT.SSL.CLIENT_CERT.ISSUER.COUNT However, if you change the value to the following, the returned count is 9: CLIENT.SSL.CLIENT_CERT.ISSUER.IGNORE_EMPTY_ELEMENTS.COUNT
<certificate>.AUTH_KEYID	Returns a string that contains the Authority Key Identifier extension of the X.509 V3 certificate.
<certificate>.AUTH_KEYID.CERT_SERIALNUMBER	Returns the SerialNumber field of the Authority Key Identifier as a blob.
<certificate>.AUTH_KEYID.EXISTS	Returns a Boolean TRUE if the certificate contains an Authority Key Identifier extension.
<certificate>.AUTH_KEYID.ISSUER_NAME	Returns the Issuer Distinguished Name in the certificate as a name-value list. An equals sign (“=”) is the delimiter f value, and the slash (“/”) is the delimiter that separates the name-value pairs. Following is an example: /C=US/O=myCompany/OU=www.mycompany.com/CN=www.mycompany.com/emailAddress=myuser
<certificate>.AUTH_KEYID.ISSUER_NAME.IGNORE_EMPTY_ELEMENTS	Returns the Issuer Distinguished Name in the certificate as a name-value list and ignores the empty elements in t For example, the following name-value list has an empty element following “a=10”: a=10;;b=11; ;c=89 The element following b=11 is not considered an empty element.
<certificate>.AUTH_KEYID.KEYID	Returns the keyIdentifier field of the Authority Key Identifier as a blob.
<certificate>.CERT_POLICY	Returns a string that contains the client certificate policy. Note that this represents a sequence of certificate po
<certificate>.KEY_USAGE(string)	Returns a Boolean value to indicate whether the specified key usage extension bit value in the X.509 certificate is set. The string argument specifies which bit is checked. Following are valid arguments: • DIGITAL_SIGNATURE. Returns TRUE if the digital signature bit is set; otherwise, it returns FALSE. • NONREPUDIATION. Returns TRUE if the nonrepudiation bit is set; otherwise, it returns FALSE. • KEYENCIPHERMENT. Returns TRUE if the key encipherment bit is set; otherwise, it returns FALSE.

SSL Certificate Operation	Description
	• DATAENCIPHERMENT. Returns TRUE if the data encipherment bit is set; otherwise, it returns FALSE. • KEYAGREEMENT. Returns TRUE if the key agreement bit is set; otherwise, it returns FALSE. • KEYCERTSIGN. Returns TRUE if the key cert sign bit is set; otherwise, it returns FALSE. • CRLSIGN. Returns TRUE if the CRL bit is set; otherwise, it returns FALSE. • ENCIPHERONLY. Returns TRUE if the encipher only bit is set; otherwise, it returns FALSE. • DECIPHERONLY. Returns TRUE if the decipher only bit is set; otherwise, it returns FALSE.
<certificate>.PK_ALGORITHM	Returns the name of the public key algorithm used by the certificate.
<certificate>.PK_SIZE	Returns the size of the public key used in the certificate.
<certificate>.SERIALNUMBER	Returns the serial number of the client certificate. If this is a non-SSL transaction or there is an error in the certifi returns an empty string.
<certificate>.SIGNATURE_ALGORITHM	Returns the name of the cryptographic algorithm used by the CA to sign this certificate.
<certificate>.SUBJECT	Returns the Distinguished Name of the Subject as a name-value. An equals sign (“=”) separates names and values delimits name-value pairs. Following is an example: /C=US/O=myCompany/OU=www.mycompany.com/CN=www.mycompany.com/emailAddress=myuser
<certificate>.SUBJECT.IGNORE_EMPTY_ELEMENTS	Returns the Subject as a name-value list, but ignores the empty elements in the list. For example, consider the fol Cert-Issuer: /c=in/st=kar//l=bangelore //o=mycompany/ou=sales/ /emailAddress=myuserid@mycc The following Rewrite action returns a count of 6 based on the preceding Issuer definition: sh rewrite action insert_ssl_header Name: insert_ssl Operation: insert_http_header Target:Cert-Issuer Value:CLIENT.SSL.CLIENT_CERT.ISSUER.COUNT However, if you change the value to the following, the returned count is 9: CLIENT.SSL.CLIENT_CERT.ISSUER.IGNORE_EMPTY_ELEMENTS.COUNT
<certificate>.SUBJECT_KEYID	Returns the Subject KeyID of the client certificate. If there is no Subject KeyID, this operation returns a zero-len

Default Syntax Expressions: IP and MAC Addresses, Throughput, VLAN IDs

May 26, 2015

You can use default syntax expression prefixes that return IPv4 and IPv6 addresses, MAC addresses, IP subnets, useful client and server data such as the throughput rates at the interface ports (Rx, Tx, and RxTx), and the IDs of the VLANs through which packets are received. You can then use various operators to evaluate the data that is returned by these expression prefixes.

This document includes the following details:

Updated: 2013-09-02

You can use default syntax expressions to evaluate addresses and subnets that are in Internet Protocol version 4 (IPv4) or Internet Protocol version 6 (IPv6) format. Expression prefixes for IPv6 addresses and subnets include IPv6 in the prefix. Expression prefixes for IPv4 addresses and subnets include IP in the prefix. Following is an example of an expression that identifies whether a request has originated from a particular IPv4 subnet.

```
client.ip.src.in_subnet(147.1.0.0/16)
```

Following are two examples of Rewrite policies that examine the subnet from which the packet is received and perform a rewrite action on the Host header. With these two policies configured, the rewrite action that is performed depends on the subnet in the request. These two policies evaluate IP addresses that are in the IPv4 address format.

```
add rewrite action URL1-rewrite-action replace "http.req.header(\"Host\")" "\"www.mycompany1.com\""
add rewrite policy URL1-rewrite-policy "http.req.header(\"Host\").contains(\"www.test1.com\") && client.ip.src.in_subnet(147.1.0.0/16)" URL1-rewrite-action
add rewrite action URL2-rewrite-action replace "http.req.header(\"Host\")" "\"www.mycompany2.com\""
add rewrite policy URL2-rewrite-policy "http.req.header(\"Host\").contains(\"www.test2.com\") && client.ip.src.in_subnet(10.202.0.0/16)" URL2-rewrite-action
```

Note: The preceding examples are commands that you type at the NetScaler command-line interface (CLI) and, therefore, each quotation mark must be preceded by a backslash (\). For more information, see "Configuring Default Syntax Expressions in a Policy."

Prefixes for IPV4 Addresses and IP Subnets

Updated: 2013-09-02

The following table describes prefixes that return IPv4 addresses and subnets, and segments of IPv4 addresses. You can use numeric operators and operators that are specific to IPv4 addresses with these prefixes. For more information about numeric operations, see "Basic Operations on Expression Prefixes" and "Compound Operations for Numbers."

Table 1. Prefixes That Evaluate IP and MAC Addresses

Prefix	Description
CLIENT.IP.SRC	Returns the source IP of the current packet as an IP address or as a number.
CLIENT.IP.DST	Returns the destination IP of the current packet as an IP address or as a number.
SERVER.IP.SRC	Returns the source IP of the current packet as an IP address or as a number.
SERVER.IP.DST	Returns the destination IP of the current packet as an IP address or as a number.

Operations for IPV4 Addresses

The following table describes the operators that can be used with prefixes that return an IPv4 address.

Table 2. Operations on IPV4 Addresses

Prefix	Description
<ip address>.EQ(<address>)	Returns a Boolean TRUE if the IP address value is same as the <address> argument. The following example checks whether the client's destination IP address is equal to 10.100.10.100: client.ip.dst.eq(10.100.10.100)
<ip address>.GET1. . .GET4	Returns a portion of an IP address as a numeric value. For example, if the IP address value is 10.100.200.1, the following is returned: client.ip.src.get1 Returns 10 client.ip.src.get2 returns 100

client.ip.src.get3 returns 200

Prefix	Description
<ip address>.IN_SUBNET(<subnet>)	Returns a Boolean TRUE if the <subnet> argument matches the subnet of the IP address value. For example, the following determines whether the client's destination IP address subnet is 10.100.10.100/18: client.ip.dst.eq(10.100.10.100/18)
<ip address>.SUBNET(<n>)	Returns the IP address after applying the subnet mask specified as the argument. The subnet mask can take values between 0 and 32. For example: CLIENT.IP.SRC.SUBNET(24) returns 192.168.1.0 if the IP address represented by the prefix is 192.168.1.[0-255].
<ip address>.IS_IPV6	Returns a Boolean TRUE if this is an Internet Protocol version 6 (IPv6) host for the client or server. Following is an example: client.ip.src.is_ipv6
<ip address>.MATCHES(<hostname>)	Returns a Boolean TRUE if the IP address for the host specified in <hostname> matches the current IP address. The <hostname> cannot exceed 255 characters.
<ip address>.MATCHES_LOCATION(<location>)	Returns a Boolean TRUE if the location of the IP address matches the <location> argument. The Location string can take the following form: qual1.qual2.qual3.qual4.qual5.qual6, for example: NorthAmeria.CA.* Following is an example: client.ip.src.matches_location(\"Europe.GB.17.London.*.*\")

About IPv6 Expressions

The IPv6 address format allows more flexibility than the older IPv4 format. IPv6 addresses are in the hexadecimal format (RFC 2373). In the following examples, Example 1 is an IPv6 address, Example 2 is a URL that includes the IPv6 address, and Example 3 includes the IPv6 address and a port number.

Example 1:
9901:0ab1:22a2:88a3:3333:4a4b:5555:6666
Example 2:
http://[9901:0ab1:22a2:88a3:3333:4a4b:5555:6666]/
Example 3:
https://[9901:0ab1:22a2:88a3:3333:4a4b:5555:6666]:8080/
In Example 3, the brackets separate the IP address from the port number (8080).

Note that you can only use the '+' operator to combine IPv6 expressions with other expressions. The output is a concatenation of the string values that are returned from the individual expressions. You cannot use any other arithmetic operator with an IPv6 expression. The following syntax is an example:

client.ipv6.src + server.ip.dst
For example, if the client source IPv6 address is ABCD:1234::ABCD, and the server destination IPv4 address is 10.100.10.100, the preceding expression returns "ABCD:1234::ABCD10.100.10.100".

Note that when the NetScaler appliance receives an IPv6 packet, it assigns a temporary IPv4 address from an unused IPv4 address range and changes the source address of the packet to this temporary address. At response time, the outgoing packet's source address is replaced with the original IPv6 address.

Note: You can combine an IPv6 expression with any other expression except an expression that produces a Boolean result.

Expression Prefixes for IPv6 Addresses

The IPv6 addresses that are returned by the expression prefixes in the following table can be treated as text data. For example, the prefix client.ipv6.dst returns the destination IPv6 address as a string that can be evaluated as text.

The following table describes expression prefixes that return an IPv6 address.

Table 3. IPv6 Expression Prefixes That Return Text

Prefix	Description
CLIENT.IPV6	Operates on the IPv6 address in with the current packet.
CLIENT.IPV6.DST	Returns the IPv6 address in the destination field of the IP header.
CLIENT.IPV6.SRC	Returns the IPv6 address in the source field of the IP header. Following are examples:

Prefix	Description
	client.ipv6.src.in_subnet(2007::2008/64) client.ipv6.src.get1.le(2008)
SERVER.IPV6	Operates on the IPv6 address in with the current packet.
SERVER.IPV6.DST	Returns the IPv6 address in the destination field of the IP header.
SERVER.IPV6.SRC	Returns the IPv6 address in the source field of the IP header. Following are examples: server.ipv6.src.in_subnet(2007::2008/64) server.ipv6.src.get1.le(2008)

Operations for IPV6 Prefixes

The following table describes the operators that can be used with prefixes that return an IPv6 address:

Table 4. Operations That Evaluate IPv6 Addresses

IPv6 Operation	Description
<ipv6>.EQ(<IPv6_address>)	Returns a Boolean TRUE if the IP address value is same as the <IPv6_address> argument. Following is an example: client.ipv6.dst.eq(ABCD:1234::ABCD)
<ipv6>.GET1. . .GET8	Returns a segment of an IPv6 address as a number. The following example expressions retrieve segments from the ipv6 address 1000:1001:CD10:0000:0000:89AB:4567:CDEF: • client.ipv6.dst.get5 extracts 0000, which is the fifth set of bits in the address. • client.ipv6.dst.get6 extracts 89AB. • client.ipv6.dst.get7 extracts 4567. You can perform numeric operations on these segments. Note that you cannot perform numeric operations when you retrieve an entire IPv6 address. This is because expressions that return an entire IPv6 address, such as CLIENT.IPV6.SRC, return the address in text format.
<ipv6>.IN_SUBNET(<subnet>)	Returns a Boolean TRUE if the IPv6 address value is in the subnet specified by the <subnet> argument. Following is an example: client.ipv6.dst.eq(1000:1001:CD10:0000:0000:89AB:4567:CDEF/60)
<ipv6>.IS_IPV4	Returns a Boolean TRUE if this is an IPv4 client, and returns a Boolean FALSE if it is not.
<ipv6>.SUBNET(<n>)	Returns the IPv6 address after applying the subnet mask specified as the argument. The subnet mask can take values between 0 and 128. For example: CLIENT.IPV6.SRC.SUBNET(24)

A MAC address consists of colon-delimited hexadecimal values in the format ##:##:##:##:##:##, where each "#" represents either a number from 0 through 9 or a letter from A through F. Default syntax expression prefixes and operators are available for evaluating source and destination MAC addresses.

Prefixes for MAC Addresses

The following table describes prefixes that return MAC addresses.

Table 5. Prefixes That Evaluate MAC Addresses

Prefix	Description
client.ether.dstmac	Returns the MAC address in the destination field of the Ethernet header.
client.ether.srcmac	Returns the MAC address in the source field of the Ethernet header.

Operations for MAC Addresses

The following table describes the operators that can be used with prefixes that return a MAC address.

Table 6. Operations on MAC Addresses

Prefix	Description
<mac address>.EQ(<address>)	Returns a Boolean TRUE if the MAC address value is same as the <address> argument.
<mac address>.GET1. . .GET4	Returns a numeric value extracted from the segment of the MAC address that is specified in the GET operation. For example, if the MAC address is 12:34:56:78:9a:bc, the following returns 34: client.ether.dstmac.get2

The following table describes prefixes for working with numeric client and server data, including throughput, port numbers, and VLAN IDs.

Table 7. Prefixes That Evaluate Numeric Client and Server Data

Prefix	Description
client.interface.rxthroughput	Returns an integer representing the raw received traffic throughput in kilobytes per second (KBps) for the previous seven seconds.
client.interface.txthroughput	Returns an integer representing the raw transmitted traffic throughput in KBps for the previous seven seconds.
client.interface.rxtxthroughput	Returns an integer representing the raw received and transmitted traffic throughput in KBps for the previous seven seconds.
server.interface.rxthroughput	Returns an integer representing the raw received traffic throughput in KBps for the previous seven seconds.
server.interface.txthroughput	Returns an integer representing the raw transmitted traffic throughput in KBps for the previous seven seconds.
server.interface.rxtxthroughput	Returns an integer representing the raw received and transmitted traffic throughput in KBps for the previous seven seconds.
server.vlan.id	Returns a numeric ID of the VLAN through which the current packet entered the NetScaler.
client.vlan.id	Returns a numeric ID for the VLAN through which the current packet entered the NetScaler.

Default Syntax Expressions: Stream Analytics Functions

May 21, 2015

Stream Analytics expressions begin with the ANALYTICS.STREAM(<identifier_name>) prefix. The following list describes the functions that can be used with this prefix.

COLLECT_STATS

Collect statistical data from the requests that are evaluated against the policy and create a record for each request.

REQUESTS

Return the number of requests that exist for the specified record grouping. The value returned is of type unsigned long.

BANDWIDTH

Return the bandwidth statistic for the specified record grouping. The value returned is of type unsigned long.

RESPTIME

Return the response time statistic for the specified record grouping. The value returned is of type unsigned long.

CONNECTIONS

Return the number of concurrent connections that exist for the specified record grouping. The value returned is of type unsigned long.

IS_TOP(n)

Return a Boolean TRUE if the statistical value for the specified record grouping is one among the top n groups. Otherwise, return a Boolean FALSE.

CHECK_LIMIT

Return a Boolean TRUE if the statistic for the specified record grouping has hit the preconfigured limit. Otherwise, return a Boolean FALSE.

Default Syntax Expressions: DataStream

May 25, 2015

The policy infrastructure on the Citrix NetScaler appliance includes expressions that you can use to evaluate and process database server traffic when the appliance is deployed between a farm of application servers and their associated database servers.

This document includes the following details:

- Expressions for the MySQL Protocol
- Expressions for Evaluating Microsoft SQL Server Connections

The following expressions evaluate traffic associated with MySQL database servers. You can use the request-based expressions (expressions that begin with MYSQL.CLIENT and MYSQL.REQ) in policies to make request switching decisions at the content switching virtual server bind point and the response-based expressions (expressions that begin with MYSQL.RES) to evaluate server responses to user-configured health monitors.

- **MYSQL.CLIENT**. Operates on the client properties of a MySQL connection.
- **MYSQL.CLIENT.CAPABILITIES**. Returns the set of flags that the client has set in the capabilities field of the handshake initialization packet during authentication. Examples of the flags that are set are CLIENT_FOUND_ROWS, CLIENT_COMPRESS, and CLIENT_SSL.
- **MYSQL.CLIENT.CHAR_SET**. Returns the enumeration constant assigned to the character set that the client uses. The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this prefix. Following are the character set enumeration constants:
  - LATIN2_CZECH_CS
  - DEC8_SWEDISH_CI
  - CP850_GENERAL_CI
  - GREEK_GENERAL_CI
  - LATIN1_GERMAN1_CI
  - HP8_ENGLISH_CI
  - KOI8R_GENERAL_CI
  - LATIN1_SWEDISH_CI
  - LATIN2_GENERAL_CI
  - SWE7_SWEDISH_CI
  - ASCII_GENERAL_CI
  - CP1251_BULGARIAN_CI
  - LATIN1_DANISH_CI
  - HEBREW_GENERAL_CI
  - LATIN7_ESTONIAN_CS
  - LATIN2_HUNGARIAN_CI
  - KOI8U_GENERAL_CI
  - CP1251_UKRAINIAN_CI
  - CP1250_GENERAL_CI
  - LATIN2_CROATIAN_CI
  - CP1257_LITHUANIAN_CI
  - LATIN5_TURKISH_CI

- LATIN1_GERMAN2_CI
- ARMSCII8_GENERAL_CI
- UTF8_GENERAL_CI
- CP1250_CZECH_CS
- CP866_GENERAL_CI
- KEYBCS2_GENERAL_CI
- MACCE_GENERAL_CI
- MACROMAN_GENERAL_CI
- CP852_GENERAL_CI
- LATIN7_GENERAL_CI
- LATIN7_GENERAL_CS
- MACCE_BIN
- CP1250_CROATIAN_CI
- LATIN1_BIN
- LATIN1_GENERAL_CI
- LATIN1_GENERAL_CS
- CP1251_BIN
- CP1251_GENERAL_CI
- CP1251_GENERAL_CS
- MACROMAN_BIN
- CP1256_GENERAL_CI
- CP1257_BIN
- CP1257_GENERAL_CI
- ARMSCII8_BIN
- ASCII_BIN
- CP1250_BIN
- CP1256_BIN
- CP866_BIN
- DEC8_BIN
- GREEK_BIN
- HEBREW_BIN
- HP8_BIN
- KEYBCS2_BIN
- KOI8R_BIN
- KOI8U_BIN
- LATIN2_BIN
- LATIN5_BIN
- LATIN7_BIN
- CP850_BIN
- CP852_BIN
- SWE7_BIN
- UTF8_BIN
- GEOSTD8_GENERAL_CI
- GEOSTD8_BIN
- LATIN1_SPANISH_CI
- UTF8_UNICODE_CI

- UTF8_ICELANDIC_CI
- UTF8_LATVIAN_CI
- UTF8_ROMANIAN_CI
- UTF8_SLOVENIAN_CI
- UTF8_POLISH_CI
- UTF8_ESTONIAN_CI
- UTF8_SPANISH_CI
- UTF8_SWEDISH_CI
- UTF8_TURKISH_CI
- UTF8_CZECH_CI
- UTF8_DANISH_CI
- UTF8_LITHUANIAN_CI
- UTF8_SLOVAK_CI
- UTF8_SPANISH2_CI
- UTF8_ROMAN_CI
- UTF8_PERSIAN_CI
- UTF8_ESPERANTO_CI
- UTF8_HUNGARIAN_CI
- INVAL_CHARSET

- **MYSQL.CLIENT.DATABASE**. Returns the name of the database specified in the authentication packet that the client sends to the database server. This is the databasename attribute.
- MYSQL.CLIENT.USER. Returns the user name (in the authentication packet) with which the client is attempting to connect to the database. This is the user attribute.
- **MYSQL.REQ**. Operates on a MySQL request.
- **MYSQL.REQ.COMMAND**. Identifies the enumeration constant assigned to the type of command in the request. The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this prefix. Following are the enumeration constant values:
  - SLEEP
  - QUIT
  - INIT_DB
  - QUERY
  - FIELD_LIST
  - CREATE_DB
  - DROP_DB
  - REFRESH
  - SHUTDOWN
  - STATISTICS
  - PROCESS_INFO
  - CONNECT
  - PROCESS_KILL
  - DEBUG
  - PING
  - TIME
  - DELAYED_INSERT
  - CHANGE_USER
  - BINLOG_DUMP

- TABLE_DUMP
- CONNECT_OUT
- REGISTER_SLAVE
- STMT_PREPARE
- STMT_EXECUTE
- STMT_SEND_LONG_DATA
- STMT_CLOSE
- STMT_RESET
- SET_OPTION
- STMT_FETCH

- **MYSQL.REQ.QUERY**. Identifies the query in the MySQL request.
- **MYSQL.REQ.QUERY.COMMAND**. Returns the first keyword in the MySQL query.
- **MYSQL.REQ.QUERY.SIZE**. Returns the size of the request query in integer format. The SIZE method is similar to the CONTENT_LENGTH method that returns the length of an HTTP request or response.
- **MYSQL.REQ.QUERY.TEXT**. Returns a string covering the entire query.
- **MYSQL.REQ.QUERY.TEXT(<n>)**. Returns the first n bytes of the MySQL query as a string. This is similar to HTTP.BODY(<n>).
  Parameters:

  n - Number of bytes to be returned

- **MYSQL.RES**. Operates on a MySQL response.
- **MYSQL.RES.ATLEAST_ROWS_COUNT(<i>)**. Checks whether the response has at least i number of rows and returns a Boolean TRUE or FALSE to indicate the result.
  Parameters:

  i - Number of rows

- **MYSQL.RES.ERROR**. Identifies the MySQL error object. The error object includes the error number and the error message.
- **MYSQL.RES.ERROR.MESSAGE**. Returns the error message that is retrieved from the server's error response.
- **MYSQL.RES.ERROR.NUM**. Returns the error number that is retrieved from the server's error response.
- **MYSQL.RES.ERROR.SQLSTATE**. Returns the value of the SQLSTATE field in the server's error response. The MySQL server translates error number values to SQLSTATE values.
- **MYSQL.RES.FIELD(<i>)**. Identifies the packet that corresponds to the i^{th} individual field in the server's response. Each field packet describes the properties of the associated column. The packet count (i) begins at 0.
  Parameters:

  i - Packet number

- **MYSQL.RES.FIELD(<i>).CATALOG**. Returns the catalog property of the field packet.
- **MYSQL.RES.FIELD(<i>).CHAR_SET**. Returns the character set of the column. The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this prefix.
- **MYSQL.RES.FIELD(<i>).DATATYPE**. Returns an enumeration constant that represents the data type of the column. This is the type (also called enum_field_type) attribute of the column. The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this prefix. The possible values for the various data types are:
  - DECIMAL

    - TINY
    - SHORT
    - LONG
    - FLOAT
    - DOUBLE
    - NULL
    - TIMESTAMP
    - LONGLONG
    - INT24
    - DATE
    - TIME
    - DATETIME
    - YEAR
    - NEWDATE
    - VARCHAR (new in MySQL 5.0)
    - BIT (new in MySQL 5.0)
    - NEWDECIMAL (new in MySQL 5.0)
    - ENUM
    - SET
    - TINY_BLOB
    - MEDIUM_BLOB
    - LONG_BLOB
    - BLOB
    - VAR_STRING
    - STRING
    - GEOMETRY
- **MYSQL.RES.FIELD(<i>).DB**. Returns the database identifier (db) attribute of the field packet.
- **MYSQL.RES.FIELD(<i>).DECIMALS**. Returns the number of positions after the decimal point if the type is DECIMAL or NUMERIC. This is the decimals attribute of the field packet.
- **MYSQL.RES.FIELD(<i>).FLAGS**. Returns the flags property of the field packet. Following are the possible hexadecimal flag values:
    - 0001: NOT_NULL_FLAG
    - 0002: PRI_KEY_FLAG
    - 0004: UNIQUE_KEY_FLAG
    - 0008: MULTIPLE_KEY_FLAG
    - 0010: BLOB_FLAG
    - 0020: UNSIGNED_FLAG
    - 0040: ZEROFILL_FLAG
    - 0080: BINARY_FLAG
    - 0100: ENUM_FLAG
    - 0200: AUTO_INCREMENT_FLAG
    - 0400: TIMESTAMP_FLAG
    - 0800: SET_FLAG
- **MYSQL.RES.FIELD(<i>).LENGTH**. Returns the length of the column. This is the value of the length attribute of the field packet. The value that is returned might be larger than the actual value. For example, an instance of a VARCHAR(2) column might return a value of 2 even when it contains only one character.

- **MYSQL.RES.FIELD(<i>).NAME**. Returns the column identifier (the name after the AS clause, if any). This is the name attribute of the field packet.
- **MYSQL.RES.FIELD(<i>).ORIGINAL_NAME**. Returns the original column identifier (before the AS clause, if any). This is the org_name attribute of the field packet.
- **MYSQL.RES.FIELD(<i>).ORIGINAL_TABLE**. Returns the original table identifier of the column (before the AS clause, if any). This is the org_table attribute of the field packet.
- **MYSQL.RES.FIELD(<i>).TABLE**. Returns the table identifier of the column (after the AS clause, if any). This is the table attribute of the field packet.
- **MYSQL.RES.FIELDS_COUNT**. Returns the number of field packets in the response (the field_count attribute of the OK packet).
- **MYSQL.RES.OK**. Identifies the OK packet sent by the database server.
- **MYSQL.RES.OK.AFFECTED_ROWS**. Returns the number of rows affected by an INSERT, UPDATE, or DELETE query. This is the value of the affected_rows attribute of the OK packet.
- **MYSQL.RES.OK.INSERT_ID**. Identifies the unique_id attribute of the OK packet. If an auto-increment identity is not generated by the current MySQL statement or query, the value of unique_id, and hence the value returned by the expression, is 0.
- **MYSQL.RES.OK.MESSAGE**. Returns the message property of the OK packet.
- **MYSQL.RES.OK.STATUS**. Identifies the bit string in the server_status attribute of the OK packet. Clients can use the server status to check whether the current command is a part of a running transaction. The bits in the server_status bit string correspond to the following fields (in the given order):
  - IN TRANSACTION
  - AUTO_COMMIT
  - MORE RESULTS
  - MULTI QUERY
  - BAD INDEX USED
  - NO INDEX USED
  - CURSOR EXISTS
  - LAST ROW SEEN
  - DATABASE DROPPED
  - NO BACKSLASH ESCAPES
- **MYSQL.RES.OK.WARNING_COUNT**. Returns the warning_count attribute of the OK packet.
- **MYSQL.RES.ROW(<i>)**. Identifies the packet that corresponds to the i^{th} individual row in the database server's response.
  Parameters:

  i - Row number

- **MYSQL.RES.ROW(<i>).DOUBLE_ELEM(<j>)**. Checks whether the j^{th} column of the i^{th} row of the table is NULL. Following C conventions, both indexes i and j start from 0. Therefore, row i and column j are actually the $(i+1)^{th}$ row and the $(j+1)^{th}$ column, respectively.
  Parameters:

  i - Row number

  j - Column number

- **MYSQL.RES.ROW(<i>).IS_NULL_ELEM(j)**. Checks whether the j^{th} column of the i^{th} row of the table is NULL. Following C conventions, both indexes i and j start from 0. Therefore, row i and column j are actually the $(i+1)^{th}$ row

and the $(j+1)^{th}$ column, respectively.
Parameters:

i - Row number

j - Column number

- **MYSQL.RES.ROW(<i>).NUM_ELEM(<j>)**. Returns an integer value from the j^{th} column of the i^{th} row of the table. Following C conventions, both indexes i and j start from 0. Therefore, row i and column j are actually the $(i+1)^{th}$ row and the $(j+1)^{th}$ column, respectively.
  Parameters:

  i - Row number

  j - Column number

- **MYSQL.RES.ROW(<i>).TEXT_ELEM(j)**. Returns a string from the j^{th} column of the i^{th} row of the table. Following C conventions, both indexes i and j start from 0. Therefore, row i and column j are actually the $(i+1)^{th}$ row and the $(j+1)^{th}$ column, respectively.
  Parameters:

  i - Row number

  j - Column number

- **MYSQL.RES.TYPE**. Returns an enumeration constant for the response type. Its values can be ERROR, OK, and RESULT_SET. The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this prefix.

The following expressions evaluate traffic associated with Microsoft SQL Server database servers. You can use the request-based expressions (expressions that begin with MSSQL.CLIENT and MSSQL.REQ) in policies to make request switching decisions at the content switching virtual server bind point and the response-based expressions (expressions that begin with MSSQL.RES) to evaluate server responses to user-configured health monitors.

Table 1. Expressions for Evaluating Microsoft SQL Server Connections

Expression	Description
MSSQL.CLIENT.CAPABILITIES	Returns the OptionFlags1, OptionFlags2, OptionFlags3, and TypeFlags fields of the LOGIN7 authentication packet, in that order, as a 4-byte integer. Each field is 1 byte long and specifies a set of client capabilities.
MSSQL.CLIENT.DATABASE	Returns the name of the client database. The value returned is of type text.
MSSQL.CLIENT.USER	Returns the user name with which the client authenticated. The value returned is of type text.

Expression	Description
MSSQL.REQ.COMMAND	Returns an enumeration constant that identifies the type of command in the request sent to a Microsoft SQL Server database server. The value returned is of type text. Examples of the values of the enumeration constant are QUERY, RESPONSE, RPC, and ATTENTION. The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this expression.
MSSQL.REQ.QUERY.COMMAND	Returns the first keyword in the SQL query. The value returned is of type text.
MSSQL.REQ.QUERY.SIZE	Returns the size of the SQL query in the request. The value returned is a number.
MSSQL.REQ.QUERY.TEXT	Returns the entire SQL query as a string. The value returned is of type text.
MSSQL.REQ.QUERY.TEXT(<n>)	Returns the first n bytes of the SQL query. The value returned is of type text. **Parameters:** n - Number of bytes
MSSQL.REQ.RPC.NAME	Returns the name of the procedure that is being called in a remote procedure call (RPC) request. The name is returned as a string.
MSSQL.REQ.RPC.IS_PROCID	Returns a Boolean value that indicates whether the remote procedure call (RPC) request contains a procedure ID or an RPC name. A return value of TRUE indicates that the request contains a procedure ID and a return value of FALSE indicates that the request contains an RPC name.
MSSQL.REQ.RPC.PROCID	Returns the procedure ID of the remote procedure call (RPC) request as an integer.
MSSQL.REQ.RPC.BODY Note: Not available for releases before 10.1.	Returns the body of the SQL request as a string in the form of parameters represented as "a=b" clauses separated by commas, where "a" is the RPC parameter name and "b" is its value.

Expression	Description
MSSQL.REQ.RPC.BODY(n) Note: Not available for releases before 10.1.	Returns part of the body of the SQL request as a string in the form of parameters represented as "a=b" clauses separated by commas, where "a" is the RPC parameter name and "b" is its value. Parameters are returned from only the first "n" bytes of the request, skipping the SQL header. Only complete name-value pairs are returned.
MSSQL.RES.ATLEAST_ROWS_COUNT(i)	Checks whether the response has at least i number of rows. The value returned is a Boolean TRUE or FALSE value. **Parameters:** i - Number of rows
MSSQL.RES.DONE.ROWCOUNT	Returns a count of the number of rows affected by an INSERT, UPDATE, or DELETE query. The value returned is of type unsigned long.
MSSQL.RES.DONE.STATUS	Returns the status field from the DONE token sent by a Microsoft SQL Server database server. The value returned is a number.
MSSQL.RES.ERROR.MESSAGE	Returns the error message from the ERROR token sent by a Microsoft SQL Server database server. This is the value of the MsgText field in the ERROR token. The value returned is of type text.
MSSQL.RES.ERROR.NUM	Returns the error number from the ERROR token sent by a Microsoft SQL Server database server. This is the value of the Number field in the ERROR token. The value returned is a number.
MSSQL.RES.ERROR.STATE	Returns the error state from the ERROR token sent by a Microsoft SQL Server database server. This is the value of the State field in the ERROR token. The value returned is a number.
MSSQL.RES.FIELD(<i>).DATATYPE	Returns the data type of the i^{th} field in the server response. The EQ(<m>) and NE(<m>) functions, which return Boolean values to indicate the result of a comparison, are used with this prefix. For example, the following expression returns a Boolean TRUE if the DATATYPE function returns a value of datetime for the third field in the response:

Expression	Description
	Parameters: i - Row number
MSSQL.RES.FIELD(<i>).LENGTH	Returns the maximum possible length of the i^{th} field in the server response. The value returned is a number. **Parameters:** i - Row number
MSSQL.RES.FIELD(<i>).NAME	Returns the name of the i^{th} field in the server response. The value returned is of type **text**. **Parameters:** i - Row number
MSSQL.RES.ROW(<i>).DOUBLE_ELEM(<j>)	Returns a value of type **double** from the j^{th} column of the i^{th} row of the table. If the value is not a double value, an **UNDEF** condition is raised. Following C conventions, both indexes i and j start from 0 (zero). Therefore, row i and column j are actually the $(i + 1)^{th}$ row and the $(j + 1)^{th}$ column, respectively. **Parameters:** i - Row number j - Column number
MSSQL.RES.ROW(<i>).NUM_ELEM(j)	Returns an integer value from the j^{th} column of i^{th} row of the table. If the value is not an integer value, an **UNDEF** condition is raised. Following C conventions, both indexes i and j start from 0 (zero). Therefore, row i and column j are actually the $(i + 1)^{th}$ row and the $(j + 1)^{th}$ column, respectively. **Parameters:** i - Row number j - Column number
MSSQL.RES.ROW(<i>).IS_NULL_ELEM(j)	Checks whether the j^{th} column of the i^{th} row of the table is **NULL** and returns a Boolean **TRUE** or **FALSE** to indicate the result. Following C conventions, both indexes i and j start from 0 (zero). Therefore, row i and column j are actually the $(i + 1)^{th}$

Expression	Description
	row and the (j + 1)th column, respectively. **Parameters:** i - Row number j - Column number
MSSQL.RES.ROW(<i>).TEXT_ELEM(j)	Returns a text string from the jth column of ith row of the table. Following C conventions, both indexes i and j start from 0 (zero). Therefore, row i and column j are actually the (i + 1)th row and the (j + 1)th column, respectively. **Parameters:** i - Row number j - Column number
MSSQL.RES.TYPE	Returns an enumeration constant that identifies the response type. Following are the possible return values: • ERROR • OK • RESULT_SET The EQ(<m>) and NE(<m>) operators, which return Boolean values to indicate the result of a comparison, are used with this expression.

Typecasting Data

May 14, 2012

You can extract data of one type (for example, text or an integer) from requests and responses and transform it to data of another type. For example, you can extract a string and transform the string to time format. You can also extract a string from an HTTP request body and treat it like an HTTP header or extract a value from one type of request header and insert it in a response header of a different type.

After typecasting the data, you can apply any operation that is appropriate for the new data type. For example, if you typecast text to an HTTP header, you can apply any operation that is applicable to HTTP headers to the returned value.

The following table describes various typecasting operations.

Table 1. Typecasting Functions

Function	Description
<text>.TYPECAST_LIST_T(<separator>)	Treats the text in an HTTP request or response body as a list whose elements are delimited by the character in the <separator> argument. Index values in the list that is created start with zero (0). Text mode settings have no effect on the separator. For example, even if you set the text mode to IGNORECASE, and the separator is the letter "p," an uppercase "P" is not treated as a separator. The following example creates a Rewrite action that constructs a list from an HTTP request body and extracts the fourth item in the list: add rewrite action myreplace_action REPLACE 'http.req.body(100)' 'http.req.body(100).typecast_list_t('?').get(4)' set rewrite policy myreplace_policy -action myreplace_action This policy returns the string "fourth item" from the following request: GET?first item?second item?third item?fourth item? The following example extracts the fourth-from-last item from the list. add rewrite action myreplace_action1 REPLACE 'http.req.body(100)' 'http.req.body(100).typecast_list_t('?').get_reverse(4)' set rewrite policy myreplace_policy1 -action myreplace_action1 This policy returns the string "first item" from the following request: GET?first item?second item?third item?fourth item.
<text>.TYPECAST_ NVLIST_T(<separator>, <delimiter>) or text.TYPECAST_ NVLIST_T(<separator>, <delimiter>, <quote>)	Treats the text as a name-value list. The <separator> argument identifies the character and separates the name and the value. The <delimiter> argument identifies the character that separates each name-value pair. The <quote> character is required when typecasting text into a name-value list that supports quoted strings. Any delimiters that appear within the quoted string are ignored. The text mode has no effect on the delimiters. For example, if the current text mode is IGNORECASE and you specify "p" as the delimiter, an uppercase "P" is not treated as a delimiter. For example, the following policy counts the number of name-value pairs and inserts the result in a header named name-value-count: add rewrite action mycount_action insert_http_header name-value-count 'http.req.header("Cookie").typecast_nvlist_t('=',';').count' set rewrite policy mycount_policy -action mycount_action This policy can extract a count of arguments in Cookie headers and insert the count in a name-value-count header: Cookie: name=name1; rank=rank1

Function	Description
<text>.TYPECAST_TIME_T	Treats the designated text as a date string. The following formats are supported: • RFC822: Sun, 06 Nov 1994 08:49:37 GMT • RFC850: Sunday, 06-Nov-94 08:49:37 GMT • ASCII TIME: Sun Nov 6 08:49:37 1994 • HTTP Set-Cookie Expiry date: Sun, 06-Nov-1994 08:49:37 GMT For example, the following policy converts the string to a time value and then extracts the day. This policy matches all requests that have a day value lesser than or equal to 10. Add rewrite policy mytime_policy "http.req.body(100) .typecast_time_t.day.le(10)" mytime_action bind rewrite global mytime_policy 100
<numeric string>.TYPECAST_IP_ ADDRESS_T	Treats a numeric string as an IP address. For example, the following policy matches HTTP requests that contains Cookie headers with a value of: 12.34.56.78\r\n. set rewrite policy ip_check_policy -rule 'http.req.cookie .value("ip").typecast_ip_address_t.eq(12.34.56.78)' bind rewrite global ip_check_policy 200 -type req_default
<numeric string>.TYPECAST_IPV6_ADDRESS_T	Treats a string as an IPv6 address in the following format: 0000:0000:CD00:0000:0000:00AB:0000:CDEF
<text>.TYPECAST_HTTP_ URL_T	Treats the designated text as the URL in the first line of an HTTP request header. The supported format is [<protocol>://<hostname>]<path>?<query>, and the text mode is set to URLENCODED by default. For example, the following policy replaces a URL-encoded part of a string in an HTTP header named Test. add rewrite action replace_header_string replace "http.req.header(\"Test\").typecast_http_url_t.path .before_str(\"123\").after_str(\"ABC\")" "\"string\"" add rewrite policy rewrite_test_header_policy true replace_header_string bind rewrite global rewrite_test_header_policy 1 END -type res_override Consider the following header: Test: ABC%12123\r\n This policy would replace the preceding header with the value ABC%string123\r\n.
<text>.TYPECAST_HTTP_ HOSTNAME_T	Provides operations for parsing an HTTP host name as it appears in HTTP data. The format for a host name is abc.def.com:8080.
<text>.TYPECAST_HTTP_ METHOD_T	Converts text to an HTTP method. For example, the following policy matches any HTTP request that contains a Host header with a value equal to POST: Add rewrite policy method_policy "http.req.header(\"Host\") .typecast_http_method_t.eq(POST)" act1
<text>.TYPECAST_DNS_ DOMAIN_T	Enables the designated text to be parsed like a DNS domain name in the format ab.def.com.
<text>.TYPECAST_HTTP_ HEADER_T("<name>")	Converts the designated text to a multi-line HTTP header that you specify in a <name> argument.

Function	Description
	For example, the following expression converts “MyHeader” to “InHeader”: http.req.header("MyHeader").typecast_http_header_t("InHeader") Typically, text operations that you specify in this type of expression apply to only the last line of this header, with some exceptions. For example, the CONTAINS operation operates on values in all the lines in instances of this header type.
<text>.TYPECAST_COOKIE_T	Treats the designated text as an HTTP cookie as it appears in a Set-Cookie or Set-Cookie2 header. You can apply name-value list operations as well as text operations to the designated text. For example, you can designate equals (=) as the name-value delimiter and the semicolon (;) as the list element delimiter. If you apply name-value list operations, the list is parsed as if IGNORE_EMPTY_ELEMENTS were in effect. Each cookie begins with a cookie-name=cookie-value pair, optionally followed by attribute-value pairs that are separated by a semicolon, as follows: cookie1=value1;version=n.n;value;domain=value;path=value If the same attribute appears more than once in a cookie, the value for the first instance of the attribute is returned.
<number>.TYPECAST_DOUBLE_AT	Transforms the number to a value of data type double.
<number>.TYPECAST_IP_ADDRESS_AT	Converts the number to an IP address.
<number>.TYPECAST_TIME_AT	Converts the number to time format.
<number>.TYPECAST_TIME_AT.BETWEEN(<time1>, <time2>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> is between the lower and upper time value arguments <time1> and <time2>. The following are prerequisites for this function: • Both the lower and upper time arguments must be fully specified. For example, GMT 1995 Jan is fully specified. But GMT Jan, GMT 1995 20 and GMT Jan Mon_2 are not fully specified. • Both arguments must be either GMT or Local. • The day of the week must not be present in either argument. However, the day of the month can be specified as the first, second, third, or fourth weekday of the month (example Wed_3 is the third Wednesday of the month). • The upper time argument, <time2>, must be bigger than the lower time argument, <time1>. The following examples assume that the current time value is GMT 2005 May 1 10h 15m 30s and that the day is the first Sunday of the month of May in 2005. The result of the evaluation is given after each example. BETWEEN(GMT 2004, GMT 2006): TRUE BETWEEN(GMT 2004 Jan, GMT 2006 Nov): TRUE BETWEEN(GMT 2004 Jan, GMT 2006): TRUE BETWEEN(GMT 2005 May Sun_1, GMT 2005 May Sun_3): TRUE BETWEEN(GMT 2005 May 1, GMT May 2005 1): TRUE BETWEEN(LOCAL 2005 May 1, LOCAL May 2005 1): The result depends on the NetScaler system's timezone. Parameters: <time1> - Lower time value <time2> - Upper time value
<number>.TYPECAST_TIME_AT.DAY	Extracts the day of the month from the current system time and returns the value as a number that corresponds to the day of the month. The returned value ranges from 1 to 31.

Function	Description
<number>.TYPECAST_TIME_AT.EQ(<t>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> is equal to the time value argument <t>. The following examples assume that the current time value is GMT 2005 May 1 10h 15m 30s and that the day is the 1st Sunday of the month of May in 2005. The result of the evaluation is given after each example. EQ(GMT 2005): TRUE EQ(GMT 2005 Dec): FALSE EQ(Local 2005 May): TRUE or FALSE, depending on the time zone. EQ(GMT 10h): TRUE EQ(GMT 10h 30s): TRUE EQ(GMT May 10h): TRUE EQ(GMT Sun): TRUE EQ(GMT May Sun_1): TRUE Parameters: <t> - Time
<number>.TYPECAST_TIME_AT.GE(<t>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> is greater than or equal to the time value argument <t>. The following examples assume that the current time value is GMT 2005 May 1 10h 15m 30s and that the day is the 1st Sunday of the month of May in 2005. The result of the evaluation is given after each example. GE(GMT 2004): TRUE GE(GMT 2005 Jan): TRUE GE(Local 2005 May): TRUE or FALSE, depending on the time zone. GE(GMT 8h): TRUE GE(GMT 30m): FALSE GE(GMT May 10h): TRUE GE(GMT May 10h 0m): TRUE GE(GMT Sun): TRUE GE(GMT May Sun_1): TRUE Parameters: <t> - Time
<number>.TYPECAST_TIME_AT.GT(<t>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> is greater than the time value argument <t>. The following examples assume that the current time value is GMT 2005 May 1 10h 15m 30s and that the day is the 1st Sunday of the month of May in 2005. The result of the evaluation is given after each example. GT(GMT 2004): TRUE GT(GMT 2005 Jan): TRUE GT(Local 2005 May): TRUE or FALSE, depending on the time zone. GT(GMT 8h): TRUE GT(GMT 30m): FALSE GT(GMT May 10h): FALSE GT(GMT May 10h 0m): TRUE GT(GMT Sun): FALSE GT(GMT May Sun_1): FALSE Parameters: <t> - Time
<number>.TYPECAST_TIME_AT.HOURS	Extracts the hour from the current system time and returns the corresponding value as an integer that can range from 0 to 23.

Function	Description
<number>.TYPECAST_TIME_AT.LE(<t>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> is lesser than or equal to the time value argument <t>. The following examples assume that the current time value is GMT 2005 May 1 10h 15m 30s and that the day is the 1st Sunday of the month of May in 2005. The result of the evaluation is given after each example. LE(GMT 2006): TRUE LE(GMT 2005 Dec): TRUE LE(Local 2005 May): TRUE or FALSE, depending on the time zone. LE(GMT 8h): FALSE LE(GMT 30m): TRUE LE(GMT May 10h): TRUE LE(GMT Jun 11h): TRUE LE(GMT Wed): TRUE LE(GMT May Sun_1): TRUE Parameters: <t> - Time
<number>.TYPECAST_TIME_AT.LT(<t>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> is lesser than the time value argument <t>. The following examples assume that the current time value is GMT 2005 May 1 10h 15m 30s and that the day is the 1st Sunday of the month of May in 2005. The result of the evaluation is given after each example. LT(GMT 2006): TRUE LT(GMT 2005 Dec): TRUE LT(Local 2005 May): TRUE or FALSE, depending on the time zone. LT(GMT 8h): FALSE LT(GMT 30m): TRUE LT(GMT May 10h): FALSE LT(GMT Jun 11h): TRUE LT(GMT Wed): TRUE LT(GMT May Sun_1): FALSE Parameters: <t> - Time
<number>.TYPECAST_TIME_AT.MINUTES	Extracts the minute from the current system time and returns the value as an integer that can range from 0 to 59.
<number>.TYPECAST_TIME_AT.MONTH	Extracts the month from the current system time and returns the value as an integer that can range from 1 (January) to 12 (December).
<number>.TYPECAST_TIME_AT.RELATIVE_BOOT	Calculates the number of seconds that have elapsed after the most recent reboot or the number of seconds to the next scheduled reboot, depending on which is closer to the current time, and returns an integer. If the closest boot time is in the past, the integer is negative. If the closest boot time is in the future (scheduled reboot time), the integer is positive.
<number>.TYPECAST_TIME_AT.RELATIVE_NOW	Calculates the number of seconds between the current system time and the specified time, and returns the value as an integer. If the designated time is in the past, the integer is negative. If it is in the future, the integer is positive.
<number>.TYPECAST_TIME_AT.SECONDS	Extracts the seconds from the current system time and returns the value as an integer that can range from 0 to 59.
<number>.TYPECAST_TIME_AT.WEEKDAY	Returns an integer that corresponds to the day of the week; 0 for Sunday and 6 for Saturday.

Function	Description
<number>.TYPECAST_TIME_AT.WITHIN(<time1>, <time2>)	Returns a Boolean value (TRUE or FALSE) that indicates whether the time value designated by <number> lies within all the ranges defined by lower and upper time value arguments <time1> and <time2>. If an element of time such as the day or the hour is left unspecified in the lower argument, <time1>, then it is assumed to have the lowest value possible for its range. If an element is left unspecified in the upper argument , <time2>, then it is assumed to have the highest value possible for its range. If the year is specified in one of the arguments, then it must be specified in the other argument as well. Following are the ranges for different elements of time: • month: 1-12 • day: 1-31 • weekday: 0-6 • hour: 0-23 • minutes: 0-59 • seconds: 0-59. Each element of time in the lower time value argument defines a range in combination with the corresponding element in the upper time value argument. For the result to be TRUE, each element of time in the time value designated by <number> must lie in the corresponding range specified by the lower and upper arguments. The following examples assume that the current time value is GMT 2005 May 10 10h 15m 30s.and that the day is the second Tuesday of the month. The result of the evaluation is given after each example. WITHIN(GMT 2004, GMT 2006): TRUE WITHIN(GMT 2004 Jan, GMT 2006 Mar): FALSE (May doesn't fall in the Jan-Mar range.) WITHIN(GMT Feb, GMT): TRUE (May falls in the Feb-Dec range.) WITHIN(GMT Sun_1, GMT Sun_3): TRUE (2nd Tuesday lies within 1st Sunday and the 3rd Sunday.) WITHIN(GMT 2005 May 1 10h, GMT May 2005 1 17h): TRUE WITHIN(LOCAL 2005 May 1, LOCAL May 2005 1): The result depends on the NetScaler system's timezone. Parameters: <time1> - Lower time value <time2> - Upper time value
<number>.TYPECAST_TIME_AT.YEAR	Extracts the year from the current system time and returns the value as a four-digit integer.
<prefix>.TYPECAST_NUM_T(<type>)	Casts numeric string data to a signed 32-bit number. The argument <type> can be one of the following: • DECIMAL. Treat the string as a decimal number and cast to a signed 32-bit number. • HEX. Treat the string as a hexadecimal number and cast to a signed 32-bit number. • DECIMAL_PREFIX. Consider the part of the string up to the first occurrence of a character that is not a valid decimal character and cast to a signed 32-bit number. • HEX_PREFIX. Consider the part of the string up to the first occurrence of a character that is not a valid hexadecimal character and cast to a signed 32-bit number. For example, the following policy extracts a numeric portion of a query string, adds 4 to the number, and inserts an HTTP header named Company with the resulting decimal value. add rewrite action myadd_action insert_http_header Company "http.req.url.query.typecast_num_t(decimal).add(4)" add rewrite policy myadd_policy true myadd_action

Function	bind rewrite global myadd_policy 300 END -type RES_DEFAULT Description
	For example, this policy would extract “4444” from the following URL stub: /test/file.html?4444 The action that is associated with the policy would insert the following HTTP response header: Company: 4448\r\n
<prefix>.TYPECAST_NUM_AT	Casts a number of any data type to a number of data type integer.
<prefix>.TYPECAST_DOUBLE_AT	Casts a number of any data type to a number of data type double.
<prefix>.TYPECAST_UNSIGNED_LONG_AT	Casts a number of any data type to a number of data type unsigned long.
<prefix>.TYPECAST_NUM_T(<type>,<default>)	Casts string data to a signed 32-bit number. If the typecasting operation raises an undefined (UNDEF) condition, the function returns the value specified for default. The type argument takes the values specified for TYPECAST_NUM_T(<type>).
<prefix>.TYPECAST_UNSIGNED_LONG_T(<type>)	Casts string data to data of type unsigned long. The argument can be one of the following: • DECIMAL. Treat the string as a decimal number and cast to unsigned long. • HEX. Treat the string as a hexadecimal number and cast to unsigned long. • DECIMAL_PREFIX. Consider the part of the string up to the first occurrence of a character that is not a valid decimal character and cast to unsigned long. • HEX_PREFIX. Consider the part of the string up to the first occurrence of a character that is not a valid hexadecimal character and cast to unsigned long.
<prefix>.TYPECAST_UNSIGNED_LONG_T(<type>, <default>)	Casts string data to data of type unsigned long. If the typecasting operation raises an undefined (UNDEF) condition, the function returns the value specified for default. The type argument takes the values specified for TYPECAST_UNSIGNED_LONG_T(<type>).

Regular Expressions

Sep 02, 2013

When you want to perform string matching operations that are more complex than the operations that you perform with the CONTAINS("<string>") or EQ("<string>") operators, you use regular expressions. The policy infrastructure on the Citrix® NetScaler® appliance includes operators to which you can pass regular expressions as arguments for text matching. The names of the operators that work with regular expressions include the string REGEX. The regular expressions that you pass as arguments must conform to the regular expression syntax that is described in "http://www.pcre.org/pcre.txt." You can learn more about regular expressions at "http://www.regular-expressions.info/quickstart.html" and at "http://www.silverstones.com/thebat/Regex.html."

The target text for an operator that works with regular expressions can be either text or the value of an HTTP header. Following is the format of a default syntax expression that uses a regular expression operator to operate on text:

```
<text>.<regex_operator>(re<delimiter><regex_pattern><delimiter>)
```

The string <text> represents the default syntax expression prefix that identifies a text string in a packet (for example, HTTP.REQ.URL). The string <regex_operator> represents the regular expression operator. The regular expression always begins with the string re. A pair of matching delimiters, represented by <delimiter>, enclose the string <regex_pattern>, which represents the regular expression.

The following example expression checks whether the URL in an HTTP packet contains the string *.jpeg (where * is a wildcard) and returns a Boolean TRUE or FALSE to indicate the result. The regular expression is enclosed within a pair of slash marks (/), which act as delimiters.

```
http.req.url.regex_match(re/*.jpeg/)
```

Regular expression operators can be combined to define or refine the scope of a search. For example, `<text>.AFTER_REGEX(re/regex_pattern1/).BEFORE_REGEX(re/regex_pattern2/)` specifies that the target for string matching is the text between the patterns regex_pattern1 and regex_pattern2. You can use a text operator on the scope that is defined by the regular expression operators. For example, you can use the CONTAINS("<string>") operator to check whether the defined scope contains the string abc:

```
<text>.AFTER_REGEX(re/regex_pattern1/).BEFORE_REGEX(re/regex_pattern2/).CONTAINS("abc")
```

Note: The process of evaluating a regular expression inherently takes more time than that for an operator such as CONTAINS("<string>") or EQ("<string>"), which work with simple string arguments. You should use regular expressions only if your requirement is beyond the scope of other operators.

Basic Characteristics of Regular Expressions

Jul 10, 2013

Following are notable characteristics of regular expressions as defined on the NetScaler appliance:

- A regular expression always begins with the string “re” followed by a pair of delimiting characters (called delimiters) that enclose the regular expression that you want to use.
  For example, re#<regex_pattern># uses the number sign (#) as a delimiter.
- A regular expression cannot exceed 1499 characters.
- Digit matching can be done by using the string \d (a backslash followed by d).
- White space can be represented by using \s (a backslash followed by s).
- A regular expression can contain white spaces.

Following are the differences between the NetScaler syntax and the PCRE syntax:

- The NetScaler does not allow back references in regular expressions.
- You should not use recursive regular expressions.
- The dot meta-character also matches the newline character.
- Unicode is not supported.
- The operation SET_TEXT_MODE(IGNORECASE) overrides the (?i) internal option in the regular expression.

Operations for Regular Expressions

Jul 10, 2013

The following table describes the operators that work with regular expressions. The operation performed by a regular expression operator in a given default syntax expression depends on whether the expression prefix identifies text or HTTP headers. Operations that evaluate headers override any text-based operations for all instances of the specified header type. When you use an operator, replace <text> with the default syntax expression prefix that you want to configure for identifying text.

Table 1. Default Syntax Expression Operators That Work with Regular Expressions

Regular Expression Operation	Description
<text>.BEFORE_REGEX(<regular expression>)	Selects the text that precedes the string that matches the <regular expression> argument. If the regular expression does not match any data in the target, the expression returns a text object of length 0. The following expression selects the string "text" from "text/plain". http.res.header("content-type").before_regex(re#/#)
<text>.AFTER_REGEX(<regular expression>)	Selects the text that follows the string that matches the <regular expression> argument. If the regular expression does not match any text in the target, the expression returns a text object of length 0. The following expression extracts "Example" from "myExample": http.req.header("etag").after_regex(re/my/)
<text>.REGEX_SELECT(<regular expression>)	Selects a string that matches the <regular expression> argument. If the regular expression does not match the target, a text object of length 0 is returned. The following example extracts the string "NS-CACHE-9.0: 90" from a Via header: http.req.header("via").regex_select(re!NS-CACHE-\d\.\d:\s*\d{1,3}!)
<text>.REGEX_MATCH(<regular expression>)	Returns TRUE if the target matches a <regular expression> argument of up to 1499 characters. The regular expression must be of the following format: re<delimiter>regular expression< delimiter> Both delimiters must be the same. Additionally, the regular expression must conform to the Perl-compatible (PCRE) regular expression library syntax. For more information, go to http://www.pcre.org/pcre.txt. In particular, see the pcrepattern man page. However note the following:

Regular Expression Operation	Description
	pcrepattern man page. However, note the following: • Back-references are not allowed. • Recursive regular expressions are not recommended. • The dot metacharacter also matches the newline character. • The Unicode character set is not supported. • SET_TEXT_MODE(IGNORECASE) overrides the (?i) internal option specified in the regular expression. The following are examples: http.req.hostname.regex_match(re/[[:alpha:]]+(abc){2,3}/) http.req.url.set_text_mode(urlencoded).regex_match(re#(a*b+c*)#) The following example matches ab and aB: http.req.url.regex_match(re/a(?i)b/) The following example matches ab, aB, Ab and AB: http.req.url.set_text_mode(ignorecase).regex_match(re/ab/) The following example performs a case-insensitive, multiline match in which the dot meta-character also matches a newline character: http.req.body.regex_match(re/(?ixm) (^ab (.*) cd$) /)

Configuring Classic Policies and Expressions

May 25, 2015

Some NetScaler features use classic policies and classic expressions. As with default syntax policies, classic policies can be either global or specific to a virtual server. However, to a certain extent, the configuration method and bind points for classic policies are different from those of default syntax policies. As with default syntax expressions, you can configure named expressions and use a named expression in multiple classic policies.

The following table summarizes NetScaler features that can be configured by using classic policies.

Table 1. Policy Type and Bind Points for Policies in Features That Use Classic Policies

Feature	Virtual Servers	Supported Policies	Policy Bind Points	How You Use the Policies
System features, Authentication	None	Authentication policies	Global	For the Authentication feature, policies contain authentication schemes for different authentication methods. For example, you can configure LDAP and certificate-based authentication schemes.
SSL	None	SSL policies	• Global • Load Balancing virtual server	To determine when to apply an encryption function and add certificate information to clear text. To provide end-to-end security. After a message is decrypted, the SSL feature re-encrypts clear text and uses SSL to communicate with back-end Web servers.
Content Switching (Can use either classic or default syntax policies, but not both)	Content Switching virtual server	Content Switching policies	• Content Switching virtual server • Cache Redirection virtual server	To determine what server or group of servers is responsible for serving responses, based on characteristics of an incoming request. Request characteristics include device type, language, cookies, HTTP method, content type and associated cache server.
Compression	None	HTTP Compression policies	• Global • Content Switching virtual server • Load Balancing virtual server	To determine what type of HTTP traffic is compressed.

Feature	Virtual Servers	Supported Policies	Policy Bind Points	How You Use the Policies
			• SSL Offload virtual server • Service	
Protection features, Filter	None	Content Filtering policies	• Global • Content Switching virtual server • Load Balancing virtual server • SSL Offload virtual server • Service	To configure the behavior of the filter function.
Protection features, SureConnect	None	SureConnect policies	• Load Balancing virtual server • SSL Offload virtual server • Service	To configure the behavior of the SureConnect function.
Protection features, Priority Queuing	None	Priority Queuing policies	• Load Balancing virtual server • SSL Offload virtual server	To configure the behavior of the Priority Queuing function.
HTML Injection	None	HTML Injection Policies	• Global • Load Balancing virtual server • Content Switching virtual server • SSL Offload virtual server	To enable the NetScaler to insert text or scripts into an HTTP response that it serves to a client.
AAA - Traffic Management	None	Authentication, Authorization, Auditing, and Session policies	• Authentication virtual server (authentication, session, and auditing policies) • Load Balancing or Content Switching virtual server (authorization	To configure rules for user access to specific sessions and auditing of user access.

Feature	Virtual Servers	Supported Policies	Policy Bind Points	How You Use the Policies
			and auditing policies) • Global (session and audit policies) • AAA group or user (session, auditing, and authorization policies)	
Cache Redirection	Cache Redirection virtual server	Cache Redirection policies Map policies	Cache Redirection virtual server	To determine whether HTTP responses are served from a cache or an origin server.
Application firewall	None	Application firewall policies	Global	To identify characteristics of traffic and data that should or should not be admitted through the firewall.
NetScaler Gateway	VPN server	Pre-Authentication policies	• AAA Global • VPN vserver	To determine how the NetScaler Gateway performs authentication, authorization, auditing, and other functions, and to define rewrite rules for general Web access using the NetScaler Gateway.
		Authentication policies	• System Global • AAA Global • VPN vserver	
		Auditing policies	• User • User group • VPN vserver	
		Session policies	• VPN Global • User • User Group • VPN vserver	
		Authorization policies	• User • User Group	
		Traffic policies	• VPN Global • User	

Feature	Virtual Servers	Supported Policies	Policy Bind Points	How You Use the Policies
		TCP Compression policies	• User Group • VPN vserver VPN Global	

Configuring a Classic Policy

Oct 29, 2013

You can configure classic policies and classic expressions by using either the configuration utility or the command-line interface. A policy rule cannot exceed 1,499 characters. When configuring the policy rule, you can use named classic expressions. For more information about named expressions, see "Creating Named Classic Expressions." After configuring the policy, you bind it either globally or to a virtual server.

Note that there are small variations in the policy configuration methods for various NetScaler features.

Note: You can embed a classic expression in a default syntax expression by using the syntax SYS.EVAL_CLASSIC_EXPR**(classic_expression)**, specifying the classic_expression as the argument.

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add cmp policy <name> -rule <expression> -action <action>
- show cmp policy [<policyName>]

Example

The following commands first create a compression action and then create a compression policy that applies the action:

```
> add cmp action cmp-act-compress compress
 Done
> show cmp action cmp-act-compress
1)     Name: cmp-act-compress  Compression Type: compress
 Done
> add cmp pol cmp-pol-compress -rule ExpCheckIp -resAction cmp-act-compress
 Done
> show cmp pol cmp-pol-compress
1)     Name: cmp-pol-compress  Rule: ExpCheckIp
       Response action: cmp-act-compress      Hits: 0
 Done
>
```

1. In the navigation pane, expand the feature for which you want to configure a policy and, depending on the feature, do the following:
   - For Content Switching, Cache Redirection, and the application firewall, click Policies.
   - For SSL, click Policies, and then in the details pane, click the Policies tab.
   - For System Authentication, click Authentication, and then in the details pane, click the Policies tab.
   - For Filter, SureConnect, and Priority Queuing, expand Protection Features, select the desired function, and then in the details pane, click the Policies tab.
   - For the NetScaler Gateway, expand NetScaler Gateway, expand Policies, select the desired function, and then in the details pane, click the Policies tab.
2. For most features, click the Add button.
3. In the Create <feature name> Policy dialog box, in the Name* text box, enter a name for the policy.

Note: Note: You must begin a policy name with a letter or underscore. A policy name can consist of 1 to 31 characters, including letters, numbers, hyphen (-), period (.), pound sign (#), space ( ), and underscore (_).

4. For most features, you associate an action or a profile. For example, you may be required to select an action, or, in the case of an NetScaler Gateway or application firewall policy, you select a profile to associate with the policy. A profile is a set of configuration options that operate as a set of actions that are applied when the data being analyzed matches the policy rule.
5. Create an expression that describes the type of data that you want this policy to match.

   Depending on the type of policy you want to create, you can choose a predefined expression, or you can create a new expression. For instructions on how to create an expression for most types of classic policies, see "Configuring a Classic Expression."

   Named expressions are predefined expressions that you can reference by name in a policy rule. For more information about named expressions, see "Creating Named Classic Expressions." For a list of all the default named expressions and a definition of each, see "Expressions Reference."

6. Click Create to create your new policy.
7. Click Close to return to the Policies screen for the type of policy you were creating.

Configuring a Classic Expression

Oct 29, 2013

Classic expressions consist of the following expression elements, listed in hierarchical order:

- **Flow Type.** Specifies whether the connection is incoming or outgoing. The flow type is REQ for incoming connections and RES for outgoing connections.
- **Protocol.** Specifies the protocol, the choices for which are HTTP, SSL, TCP, and IP.
- **Qualifier.** The protocol attribute, which depends on the selected protocol.
- **Operator.** The type of test you want to perform on the connection data. Your choice of operator depends upon the connection information you are testing. If the connection information you are testing is text, you use text operators. If it is a number, you use standard numeric operators.
- **Value.** The string or number against which the connection data element—defined by the flow type, protocol, and qualifier—is tested. The value can be either a literal or an expression. The literal or expression must match the data type of the connection data element.

In a policy, classic expressions can be combined to create more complex expressions using Boolean and comparative operators.

Expression elements are parsed from left to right. The leftmost element is either REQ or RES and designates a request or a response, respectively. Successive terms define a specific connection type and a specific attribute for that connection type. Each term is separated from any preceding or following term by a period. Arguments appear in parentheses and follow the expression element to which they are passed.

The following classic expression fragment returns the client source IP for an incoming connection.

```
REQ.IP.SOURCEIP
```

The example identifies an IP address in a request. The expression element SOURCEIP designates the source IP address. This expression fragment may not be useful by itself. You can use an additional expression element, an operator, to determine whether the returned value meets specific criteria. The following expression tests whether the client IP is in the subnet 200.0.0.0/8 and returns a Boolean TRUE or FALSE:

```
REQ.IP.SOURCEIP == 200.0.0.0 -netmask 255.0.0.0
```

At the command prompt, type the following commands to set the parameters and verify the configuration:

- set appfw policy <name> -rule <expression> -action <action>
- show appfw policy <name>

Example

```
> set appfw policy GenericApplicationSSL_ 'HTTP.REQ.METHOD.EQ("get")' APPFW_DROP
 Done
> show appfw policy GenericApplicationSSL_
      Name: GenericApplicationSSL_   Rule: HTTP.REQ.METHOD.EQ("get")
      Profile: APPFW_DROP    Hits: 0
      Undef Hits: 0
      Policy is bound to following entities
```

```
    1) REQ VSERVER app_u_GenericApplicationSSLPortalPages   PRIORITY : 100
Done
```

This procedure documents the Add Expression dialog box. Depending on the feature for which you are configuring a policy, the route by which you arrive at this dialog box may be different.

1. Perform steps 1-4 in "To create a policy with classic expressions by using the configuration utility."
2. In the Add Expression dialog box, in Expression Type, click the type of expression you want to create.
3. Under Flow Type, click the down arrow and choose a flow type.

   The flow type is typically REQ or RES. The REQ option specifies that the policy applies to all incoming connections or requests. The RES option applies the policy to all outgoing connections or responses.

   For Application Firewall policies, you should leave the expression type set to General Expression, and the flow type set to REQ. The Application Firewall treats each request and response as a single paired entity, so all Application Firewall policies begin with REQ.
4. Under Protocol, click the down arrow and choose the protocol you want for your policy expression. Your choices are:
   - HTTP. Evaluates HTTP requests that are sent to a Web server. For classic expressions, HTTP includes HTTPS requests.
   - SSL. Evaluates SSL data associated with the current connection.
   - TCP. Evaluates the TCP data associated with the current connection.
   - IP. Evaluates the IP addresses associated with the current connection.
5. Under Qualifier, click the down arrow and choose a qualifier for your policy.

   The qualifier defines the type of data to be evaluated. The list of qualifiers that appears depends on which protocol you selected in step 4.

   The following list describes the qualifier choices for the HTTP protocol. For a complete list of protocols and qualifiers, see "Classic Expressions."

   The following choices appear for the HTTP protocol:

   - METHOD. Filters HTTP requests that use a particular HTTP method.
   - URL. Filters HTTP requests for a specific Web page.
   - URLQUERY. Filters HTTP requests that contain a particular query string.
   - VERSION. Filters HTTP requests on the basis of the specified HTTP protocol version.
   - HEADER. Filters on the basis of a particular HTTP header.
   - URLLEN. Filters on the basis of the length of the URL.
   - URLQUERY. Filters on the basis of the query portion of the URL.
   - URLQUERYLEN. Filters on the basis of the length of the query portion of the URL only.
6. Under Operator, click the down arrow and choose the operator for your policy expression. For a complete list of choices see the “Operators” table in "Classic Expressions." Some common operators are:

Operator	Description
==	Matches the specified value exactly or is exactly equal to the specified value.
!=	Does not match the specified value.

Operator	Description
>	Is greater than the specified value.
<	Is less than the specified value.
>=	Is greater than or equal to the specified value.
<=	Is less than or equal to the specified value.
CONTAINS	Contains the specified value.
CONTENTS	Returns the contents of the designated header, URL, or URL query.
EXISTS	The specified header or query exists.
NOTCONTAINS	Does not contain the specified value.
NOTEXISTS	The specified header or query does not exist.

7. If a Value text box appears, type a string or numeric value, as appropriate. For example, chose REQ as the Flow Type, HTTP as the Protocol, and HEADER as the qualifier, and then type the value of the header string in the Value field and the header type for which you want to match the string in the Header Name text box.
8. Click OK.
9. To create a compound expression, click Add. Note that the type of compounding that is done depends on the following choices in the Create Policy dialog box:
   - **Match Any Expression.** The expressions are in a logical OR relationship.
   - **Match All Expressions.** The expressions are in a logical AND relationship.
   - **Tabular Expressions.** Click the AND, OR, and parentheses buttons to control evaluation.
   - **Advanced Free-Form.** Enter the expressions components directly into the Expression field, and click the AND, OR, and parentheses buttons to control evaluation.

Binding a Classic Policy

Oct 29, 2013

Depending on the policy type, you can bind a classic policy either globally or to a virtual server. Policy bind points are described in the table, "Policy Type and Bind Points for Policies in Features That Use Classic Policies."

Note: You can bind a classic policy to multiple bind points.

At the command prompt, type the following commands to set the parameters and verify the configuration:

- bind cmp global <policyName> [-priority <positive_integer>]
- show cmp global

Example

```
> bind cmp global cmp-pol-compress -priority 2
 Done
> show cmp global
1)     Policy Name: cmp-pol-compress    Priority: 2
2)     Policy Name: ns_nocmp_xml_ie    Priority: 8700
3)     Policy Name: ns_nocmp_mozilla_47       Priority: 8800
4)     Policy Name: ns_cmp_mscss       Priority: 8900
5)     Policy Name: ns_cmp_msapp       Priority: 9000
6)     Policy Name: ns_cmp_content_type       Priority: 10000
 Done
>
```

At the command prompt, type the following commands to set the parameters and verify the configuration:

- bind lb vserver <name> [<targetVserver>] [-policyName <string> [-priority <positive_integer>]
- show lb vserver<name>

Example

```
> bind lb vserver lbtemp -policyName cmp-pol-compress -priority 1
 Done
> show lb vserver lbtemp
      lbtemp (10.102.29.101:80) - HTTP       Type: ADDRESS
      State: UP
      Last state change was at Tue Oct 27 06:40:38 2009 (+557 ms)
      Time since last state change: 0 days, 02:00:40.330
      Effective State: UP
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Port Rewrite : DISABLED
      No. of Bound Services :  1 (Total)      1 (Active)
      Configured Method: LEASTCONNECTION
      Current Method: Round Robin, Reason: Bound service's state changed to UP
      Group: vserver-grp
      Mode: IP
```

```
        Persistence: COOKIEINSERT (version 0)   Persistence Backup: SOURCEIP   Persistence Mask: 255.255.255.255
        Persistence Timeout: 2 min      Backup Persistence Timeout: 2 min
        Vserver IP and Port insertion: OFF
        Push: DISABLED  Push VServer:
        Push Multi Clients: NO
        Push Label Rule: none
1) http-one (10.102.29.252: 80) - HTTP State: UP       Weight: 1
        Persistence Cookie Value : NSC_wtfswfs-hsq=ffffffff096e03ed45525d5f4f58455e445a4a423660
1)      Policy : cmp-pol-compress Priority:1
 Done
>
```

Note: This procedure documents the Global Bindings dialog box. Depending on the feature for which you want to globally bind a policy, the route by which you arrive at this dialog box may be different.

1. In the navigation pane, expand the feature for which you want to globally bind a classic policy, and then locate the policy that you want to bind globally.

   Note: You cannot globally bind policies for Content Switching, Cache Redirection, SureConnect, Priority Queuing, or NetScaler Gateway Authorization.
2. In the details pane, click Global Bindings.
3. In the Bind/Unbind <feature name> Policy(s) to Global dialog box, click Insert Policy.
4. In the Policy Name column, click the name of an existing policy that you want to globally bind, or click New Policy to open the Create <feature name> Policy dialog box.
5. After you have selected the policy or created a new policy, in the Priority column, type the priority value.

   The lower the number, the sooner this policy is applied relative to other policies. For example, a policy assigned a priority of 10 is applied before a policy with a priority of 100. You can use the same priority for different policies. All features that use classic policies implement only the first policy that a connection matches, so policy priority is important for getting the results you intend.

   As a best practice, leave room to add policies by setting priorities with intervals of 50 (or 100) between each policy.
6. Click OK.

1. In the navigation pane, expand the feature that contains the virtual server to which you want to bind a classic policy (for example, if you want to bind a classic policy to a content switching virtual server, expand Traffic Management > Content Switching), and then click Virtual Servers.
2. In the details pane, select the virtual server, and then click Open.
3. In the Configure <Feature> Virtual Server dialog box, on the Policies tab, click the feature icon for the type policy that you want, and then click Insert Policy.
4. In the Policy Name column, click the name of an existing policy that you want to bind to a virtual server, or click A to open the Create <feature name> Policy dialog box.
5. After you have selected the policy or created a new policy, in the Priority column, set the priority.
   If you are binding a policy to a content switching virtual server, in the Target column, select a load balancing virtual server to which traffic that matches the policy should be sent.
6. Click OK.

Viewing Classic Policies

Oct 29, 2013

You can view classic policies by using either the configuration utility or the command line. You can view details such as the policy's name, expression, and bindings.

At the command prompt, type the following commands to view a classic policy and its binding information:

show <featureName> policy [policyName]

Example

```
> show appfw policy GenericApplicationSSL_
      Name: GenericApplicationSSL_   Rule: ns_only_get_adv
      Profile: GenericApplicationSSL_Prof1   Hits: 0
      Undef Hits: 0
      Policy is bound to following entities
      1) REQ VSERVER app_u_GenericApplicationSSLPortalPages   PRIORITY : 100
 Done
```

Note: If you omit the policy name, all policies are listed without the binding details.

1. In the navigation pane, expand the feature whose policies you want to view, (for example, if you want to view application firewall policies, expand Application Firewall), and then click Policies.
2. In the details pane, do one or more of the following:
   - To view details for a specific policy, click the policy. Details appear in the Details area of the configuration pane.
   - To view bindings for a specific policy, click the policy, and then click Show Bindings.
   - To view global bindings, click the policy, and then click Global Bindings. Note that you cannot bind a Content Switching, Cache Redirection, SureConnect, Priority Queuing, or NetScaler Gateway Authorization policy globally.

Creating Named Classic Expressions

Sep 30, 2013

A named classic expression is a classic expression that can be referenced through an assigned name. Often, you need to configure classic expressions that are large or complex and form a part of a larger compound expression. You might also configure classic expressions that you need to use frequently and in multiple compound expressions or classic policies. In these scenarios, you can create the classic expression you want, save it with a name of your choice, and then reference the expression from compound expressions or policies through its name. This saves configuration time and improves the readability of complex compound expressions. Additionally, any modifications to a named classic expression need to be made only once.

Some named expressions are built-in, and a subset of these are read-only. Built-in named expressions are divided into four categories: General, Anti-Virus, Personal Firewall, and Internet Security. General named expressions have a wide variety of uses. For example, from the General category, you can use the expressions ns_true and ns_false to specify a value of TRUE or FALSE, respectively, to be returned for all traffic. You can also identify data of a particular type (for example, HTM, DOC, or GIF files), determine whether caching headers are present, or determine whether the round trip time for packets between a client and the NetScaler is high (over 80 milliseconds).

Anti-Virus, Personal Firewall, and Internet Security named expressions test clients for the presence of a particular program and version and are used primarily in NetScaler Gateway policies.

For descriptions of the built-in named expressions, see "Classic Expressions."

Note: You cannot modify or delete built-in named expressions.

At the command prompt, type the following commands to set the parameters and verify the configuration:

- add expression <name> <value> [-comment <string>] [-clientSecurityMessage <string>]
- show expression [<name> | -type CLASSIC

Example

```
> add expression classic_ne "REQ.HTTP.URL CONTAINS www.example1.com" -comment "Checking the URL for www.example1.com"
 Done
> show expression classic_ne
1)     Name: classic_ne  Expr: REQ.HTTP.URL CONTAINS www.example1.com  Hits: 0 Type : CLASSIC
       Comment: "Checking the URL for www.example1.com"
 Done
>
```

1. In the navigation pane, expand AppExpert, expand Expressions, and then click Classic Expressions.
2. In the details pane, click Add.

   Note: Some of the built-in expressions in the Expressions list are read-only.
3. In the Create Policy Expression dialog box, specify values for the following parameters:

   - Expression Name*—name
   - Client Security Message—clientSecurityMessage
   - Comments—comment

   * A required parameter
4. To create the expression, do one of the following:
   - You can choose inputs to this expression from the Named Expressions drop-down list.
   - You can create a new expression, as described in "To add an expression for a classic policy by using the configuration utility."
5. When you are done, click Close. Verify that your new expression was created by scrolling to the bottom of the Classic Expressions list to view it.

Expressions Reference-Default Syntax Expressions

May 26, 2015

The following table is a listing of default syntax expression prefixes, with cross-references to descriptions of these prefixes and the operators that you can specify for them. Note that some prefixes can work with multiple types of operators. For example, a cookie can be parsed by using operators for text or operators for HTTP headers.

You can use any element in the following tables as a complete expression on its own, or you can use various operators to combine these expression elements with others to form more complex expressions.

Note: The Description column in the following table contains cross-references to additional information about prefix usage and applicable operators for the prefix.

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
CLIENT.ETHER	"Prefixes for MAC Addresses." "Operations for MAC Addresses."
CLIENT.ETHER.[DSTMAC | SRCMAC]	"Prefixes for MAC Addresses." "Operations for MAC Addresses."
CLIENT.INTERFACE	Designates an expression that refers to the ID of the network interface through which the current packet entered the Application Switch. See the other CLIENT.INTERFACE prefix descriptions in this table.
CLIENT.INTERFACE.ID	Extracts the ID of the network interface that received the current packet of data. See the other CLIENT.INTERFACE prefix descriptions in this table.
CLIENT.INTERFACE.ID.EQ("id")	Returns Boolean TRUE if the interface's ID matches the ID that is passed as the argument. For example: CLIENT.INTERFACE.ID.EQ("1/1") See "Booleans in Compound Expressions."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
CLIENT.INTEREACE.[RXTHROU GHPUT | RXTXTHROUGHPUT | TXTHROUGHPUT]	"Expressions for Numeric Client and Server Data." "Compound Operations for Numbers."
CLIENT.IP	Operates on the IP protocol data associated with the current packet. See the other CLIENT.IP prefixes in this table.
CLIENT.IP.DST	"Prefixes for IPV4 Addresses and IP Subnets." "Operations for IPV4 Addresses." "Compound Operations for Numbers."
CLIENT.IP.SRC	"Prefixes for IPV4 Addresses and IP Subnets." "Operations for IPV4 Addresses." "Compound Operations for Numbers."
CLIENT.IPV6	Operates on IPv6 protocol data. See the other CLIENT.IPV6 prefixes in this table.
CLIENT.IPV6.DST	"Expression Prefixes for IPv6 Addresses." "Operations for IPV6 Prefixes."
CLIENT.IPV6.SRC	"Expression Prefixes for IPv6 Addresses." "Operations for IPV6 Prefixes."
CLIENT.SSL	Operates on the SSL protocol data for the current packet. See the other CLIENT.SSL prefixes in this table.
CLIENT.SSL.CIPHER_BITS	"Prefixes for Numeric Data in SSL Certificates." "Compound Operations for Numbers."
CLIENT.SSL.CIPHER_EXPORTA BLE	"Prefixes for Text-Based SSL and Certificate Data."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
CLIENT.SSL.CLIENT_CERT	"Booleans in Compound Expressions." "Expressions for SSL Certificates." "Expressions for SSL Certificate Dates."
CLIENT.SSL.IS_SSL	"Prefixes for Text-Based SSL and Certificate Data." "Booleans in Compound Expressions."
CLIENT.SSL.VERSION	"Prefixes for Numeric Data in SSL Certificates." "Compound Operations for Numbers."
CLIENT.TCP	Operates on TCP protocol data. See the other **CLIENT.TCP** prefixes in this table.
CLIENT.TCP.[DSTPORT | MSS | SRCPORT]	"Expressions for TCP, UDP, and VLAN Data." "Compound Operations for Numbers."
CLIENT.TCP.PAYLOAD(integer)	"Expressions for TCP, UDP, and VLAN Data." "Default Syntax Expressions: Evaluating Text."
CLIENT.UDP	Operates on the UDP protocol data associated with the current packet. See the other **CLIENT.UDP** prefixes in this table.
CLIENT.UDP.DNS.DOMAIN	"Expressions for TCP, UDP, and VLAN Data." "Default Syntax Expressions: Evaluating Text."
CLIENT.UDP.DNS.DOMAIN.EQ("hostname")	"Expressions for TCP, UDP, and VLAN Data." "Booleans in Compound Expressions."
CLIENT.UDP.DNS. [IS_AAAAREC | IS_ANYREC | IS_AREC | IS_CNAMEREC | IS_MXREC | IS_NSREC | IS_PTRREC |	"Expressions for TCP, UDP, and VLAN Data."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
IS_SOAREC | IS_SRVREC]	"Booleans in Compound Expressions."
CLIENT.UDP.[DSTPORT | SRCPORT]	"Expressions for TCP, UDP, and VLAN Data." "Compound Operations for Numbers."
CLIENT.VLAN	Operates on the VLAN through which the current packet entered the NetScaler. See the other CLIENT.VLAN prefixes in this table.
CLIENT.VLAN.ID	"Expressions for TCP, UDP, and VLAN Data." "Compound Operations for Numbers."
HTTP.REQ	Operates on HTTP requests. See the other HTTP.REQ prefixes in this table.
HTTP.REQ.BODY(integer)	"Expression Prefixes for Text in HTTP Requests and Responses." "Basic Operations on Text."
HTTP.REQ.CACHE_CONTROL	"Prefixes for Cache-Control Headers." "Operations for Cache-Control Headers."
HTTP.REQ.CONTENT_LENGTH	"Expressions for Numeric HTTP Payload Data Other Than Dates." "Compound Operations for Numbers."
HTTP.REQ.COOKIE	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.REQ.DATE	"Format of Dates and Times in an Expression." "Expressions for HTTP Request and Response Dates." "Default Syntax Expressions: Evaluating

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
	Text." "Compound Operations for Numbers." "Operations for HTTP Headers."
HTTP.REQ.HEADER("header_name")	"Expression Prefixes for Text in HTTP Requests and Responses." "Prefixes for HTTP Headers." "Operations for HTTP Headers."
HTTP.REQ.FULL_HEADER("header_name")	"Prefixes for HTTP Headers." "Operations for HTTP Headers."
HTTP.REQ.HOSTNAME	"Expression Prefixes for Text in HTTP Requests and Responses."
HTTP.REQ.HOSTNAME.[DOMAIN | Server]	"Expression Prefixes for Text in HTTP Requests and Responses." "Basic Operations on Text."
HTTP.REQ.HOSTNAME.EQ("hostname")	"Expression Prefixes for Text in HTTP Requests and Responses." "Booleans in Compound Expressions." "“Basic Operations on Expression Prefixes”."
HTTP.REQ.HOSTNAME.PORT	"Expression Prefixes for Text in HTTP Requests and Responses." "Compound Operations for Numbers."
HTTP.REQ.IS_VALID	Returns TRUE if the HTTP request is properly formed. See "Booleans in Compound Expressions."
HTTP.REQ.METHOD	"Expression Prefixes for Text in HTTP Requests and Responses." "Basic Operations on Text." "Complex Operations on Text."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
HTTP.REQ.TRACKING	Returns the HTTP body tracking mechanism. See the descriptions of other HTTP.REQ.TRACKING prefixes in this table.
HTTP.REQ.TRACKING.EQ("tracking_mechanism")	Returns TRUE or FALSE. See "Booleans in Compound Expressions."
HTTP.REQ.URL	Obtains the HTTP URL object from the request and sets the text mode to URLENCODED by default. See "Expression Prefixes for Text in HTTP Requests and Responses."
HTTP.REQ.URL.[CVPN_ENCODE | HOSTNAME | HOSTNAME.DOMAIN | SERVER | PATH | PATH_AND_QUERY | PROTOCOL | QUERY | SUFFIX | VERSION]	"Expression Prefixes for Text in HTTP Requests and Responses." "Basic Operations on Text." "Complex Operations on Text."
HTTP.REQ.URL.HOSTNAME.EQ("hostname")	"Expression Prefixes for Text in HTTP Requests and Responses. "Booleans in Compound Expressions."
HTTP.REQ.URL.HOSTNAME.PORT	"Expression Prefixes for Text in HTTP Requests and Responses. "Compound Operations for Numbers."
HTTP.REQ.URL.PATH.IGNORE_ EMPTY_ELEMENTS	Ignores spaces in the data. See the table "HTTP Expression Prefixes that Return Text."
HTTP.REQ.URL.QUERY.IGNORE _EMPTY_ELEMENTS	Ignores spaces in the data. See the table "HTTP Expression Prefixes that Return Text."
HTTP.REQ.USER.IS_MEMBER_OF	"HTTP Expression Prefixes that Return Text."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
HTTP.REQ.USER.NAME	"HTTP Expression Prefixes that Return Text."
HTTP.REQ.VERSION	"Expression Prefixes for Text in HTTP Requests and Responses."
HTTP.REQ.VERSION.[MAJOR | MINOR]	Operates on the major or minor HTTP version string. See "Expression Prefixes for Text in HTTP Requests and Responses" and "Compound Operations for Numbers."
HTTP.RES	Operates on HTTP responses.
HTTP.RES.BODY(integer)	"Expression Prefixes for Text in HTTP Requests and Responses." "Basic Operations on Text." "Complex Operations on Text."
HTTP.RES.CACHE_CONTROL	"Prefixes for Cache-Control Headers." "Operations for Cache-Control Headers."
HTTP.RES.CONTENT_LENGTH	"Expression Prefixes for Text in HTTP Requests and Responses." "Operations for HTTP Headers." "Compound Operations for Numbers."
HTTP.RES.DATE	"Format of Dates and Times in an Expression." "Expressions for HTTP Request and Response Dates." "Expression Prefixes for Text in HTTP Requests and Responses." "Compound Operations for Numbers." "Operations for HTTP Headers."
HTTP.RES.HEADER("header_name")	"Expression Prefixes for Text in HTTP

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
HTTP.REQ.FULL_HEADER("header_name")	"Prefixes for HTTP Headers." "Operations for HTTP Headers."
HTTP.REQ.TXID	"Prefixes for HTTP Headers." "Operations for HTTP Headers."
HTTP.RES.IS_VALID	Returns TRUE if the HTTP response is properly formed. See "Booleans in Compound Expressions."
HTTP.RES.SET_COOKIE	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE.COOKIE("name")	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE.COOKI E.[DOMAIN | PATH | PORT]	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE.COOKIE.EXPIRES	Obtains the Expires field of the cookie as a date string. The value of the Expires attribute can be operated upon as a time object. If multiple Expires fields are present, this expression operates on the first one. If the Expires attribute is absent, a string of length zero is returned. Also see:

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text." "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE.COOKIE.PATH.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "Expression Prefixes for Text in HTTP Requests and Responses."
HTTP.RES.SET_COOKIE.COOKIE.PORT.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text."
HTTP.RES.SET_COOKIE.COOKIE.VERSION	"Prefixes for HTTP Headers." "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE.COOKIE("name",integer)[.PORT | PATH | DOMAIN | VERSION | EXPIRES]	"Prefixes for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE.COOKIE.EXPIRES	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text." "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE.EXISTS("name")	"Prefixes for HTTP Headers." "Booleans in Compound Expressions."
HTTP.RES.SET_COOKIE2	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
HTTP.RES.SET_COOKIE2.COOKIE("name")	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE2.COOKIE.[DOMAIN | PATH | PORT]	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE2.COOKIE.EXPIRES	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text." "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE2.COOKIE.PATH.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table HTTP Expression Prefixes that Return Text."
HTTP.RES.SET_COOKIE2.COOKIE.PORT.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text." See also "Default Syntax Expressions: Evaluating Text" and "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE2.COOKIE("name",integer)[.PORT | PATH | DOMAIN | VERSION | EXPIRES]	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."
HTTP.RES.SET_COOKIE2.COOKIE.DOMAIN	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
HTTP.RES.SET_COOKIE2.COOKIE.EXPIRES	"Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text." "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE2.COOKIE.VERSION	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Default Syntax Expressions: Evaluating Text." "Compound Operations for Numbers."
HTTP.RES.SET_COOKIE2.EXISTS("name")	"Prefixes for HTTP Headers." "Operations for HTTP Headers." "Booleans in Compound Expressions."
HTTP.RES.STATUS	"Expression Prefixes for Text in HTTP Requests and Responses." "Compound Operations for Numbers."
HTTP.RES.STATUS_MSG	"Expression Prefixes for Text in HTTP Requests and Responses."
HTTP.RES.TRACKING	Returns the HTTP body tracking mechanism. See the descriptions of other HTTP.REQ.TRACKING prefixes in this table.
HTTP.RES.TRACKING.EQ("tracking_method")	Returns TRUE or FALSE. See "Booleans in Compound Expressions."
HTTP.RES.TXID	" Prefixes for HTTP Headers." "Operations for HTTP Headers."
HTTP.RES.VERSION	"Expression Prefixes for Text in HTTP Requests and Responses."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
HTTP.RES.VERSION.[MAJOR | MINOR]	Operates on the major or minor HTTP version string. See "Expression Prefixes for Text in HTTP Requests and Responses" and "Compound Operations for Numbers."
SERVER	Designates an expression that refers to the server. This is the starting point for access into parameters such as Ether and SSL. See the other SERVER prefixes in this table.
SERVER.ETHER	Operates on the ethernet protocol data associated with the current packet. See the other SERVER prefixes in this table.
SERVER.ETHER.DSTMAC	"Prefixes for MAC Addresses." "Prefixes for MAC Addresses."
SERVER.INTERFACE	Designates an expression that refers to the ID of the network interface that received the current packet of data. See the other SERVER.INTERFACE prefixes in this table.
SERVER.INTERFACE.ID.EQ("id")	Returns Boolean TRUE if the interface's ID matches the ID that is passed as the argument. For example: SERVER.INTERFACE.ID.EQ("LA/1") See "Booleans in Compound Expressions."
SERVER.INTERFACE.[RXTHROUGHPUT | RXTXTHROUGHPUT | TXTHROUGHPUT]	"Expressions for Numeric Client and Server Data." "Compound Operations for Numbers."
SERVER.IP	Operates on the IP protocol data associated with the current packet. See the other SERVER.IP prefixes in this table.

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
SERVER.IP.[DST | SRC]	"Prefixes for IPV4 Addresses and IP Subnets." "Operations for IPV4 Addresses." "Compound Operations for Numbers."
SERVER.IPV6	Operates on IPv6 protocol data. See the other SERVER.IPV6 prefixes in this table.
SERVER.IPV6.DST	"Expression Prefixes for IPv6 Addresses." "Operations for IPV6 Prefixes."
SERVER.IPV6.SRC	"Expression Prefixes for IPv6 Addresses." "Operations for IPV6 Prefixes."
SERVER.TCP	Operates on TCP protocol data. See the other CLIENT.TCP prefixes in this table.
SERVER.TCP.[DSTPORT | MSS | SRCPORT]	"Expressions for TCP, UDP, and VLAN Data." "Compound Operations for Numbers."
SERVER.VLAN	Operates on the VLAN through which the current packet entered the NetScaler. See the other SERVER.VLAN prefixes in this table.
SERVER.VLAN.ID	"Expressions for TCP, UDP, and VLAN Data." "Compound Operations for Numbers."
SYS	Designates an expression that refers to the NetScaler itself, not to the client or server.. See the other SYS prefixes in this table.
SYS.EVAL_CLASSIC_EXPR(cla ssic_expression)	"Classic Expressions in Default Syntax Expressions." "Booleans in Compound Expressions."
SYS.HTTP_CALLOUT(http_callout)	"HTTP Callouts."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
SYS.CHECK_LIMIT	"Rate Limiting."
SYS.TIME	"Expressions for the NetScaler System Time." "Compound Operations for Numbers."
SYS.TIME.[BETWEEN(time1, time2) | EQ(time) | GE(time) | GT(time) | LE(time) | LT(time) | WITHIN(time1, time2)]	"Expressions for the NetScaler System Time." "Booleans in Compound Expressions." "Compound Operations for Numbers."
SYS.TIME.[DAY | HOURS | MINUTES | MONTH | RELATIVE_BOOT | RELATIVE_NOW SECONDS | WEEKDAY | YEAR]	"Expressions for the NetScaler System Time." "Compound Operations for Numbers."
SYS.RANDOM	Returns a random number between 0 and 1, inclusive of 0 but exclusive of 1.
VPN.BASEURL.[CVPN_DECODE | CVPN_ENCODE | HOSTNAME | HOSTNAME.DOMAIN | HOSTNAME.SERVER | PATH | PATH_AND_QUERY | PROTOCOL | QUERY | SUFFIX]	"Expression Prefixes for VPNs and Clientless VPNs."
VPN.BASEURL.HOSTNAME.EQ("hostname")	"Expression Prefixes for VPNs and Clientless VPNs." "Booleans in Compound Expressions."
VPN.BASEURL.HOSTNAME.PORT	"Expression Prefixes for VPNs and Clientless VPNs." "Compound Operations for Numbers."
VPN.BASEURL.PATH.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text."
VPN.BASEURL.QUERY.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text."

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
VPN.CLIENTLESS_BASEURL VPN.CLIENTLESS_BASEURL. [CVPN_DECODE | CVPN_ENCODE | HOSTNAME | HOSTNAME.DOMAIN | HOSTNAME.SERVER | PATH | PATH_AND_QUERY | PROTOCOL | QUERY | SUFFIX]	"Expression Prefixes for VPNs and Clientless VPNs." "Expression Prefixes for VPNs and Clientless VPNs."
VPN.CLIENTLESS_BASEURL.HOSTNAME.EQ("hostname")	"Expression Prefixes for VPNs and Clientless VPNs." "Booleans in Compound Expressions."
VPN.CLIENTLESS_BASEURL.HOSTNAME.PORT	"Expression Prefixes for VPNs and Clientless VPNs." "Compound Operations for Numbers."
VPN.CLIENTLESS_BASEURL.PATH.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text."
VPN.CLIENTLESS_BASEURL.QUERY.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text."
VPN.CLIENTLESS_HOSTURL	" Expression Prefixes for VPNs and Clientless VPNs."
VPN.CLIENTLESS_HOSTURL.[CVPN_DECODE | CVPN_ENCODE | HOSTNAME | HOSTNAME.DOMAIN | HOSTNAME.SERVER | PATH | PATH_AND_QUERY | PROTOCOL | QUERY | SUFFIX]	"Expression Prefixes for VPNs and Clientless VPNs."
VPN.CLIENTLESS_HOSTURL.HOSTNAME.EQ("hostname")	"Expression Prefixes for VPNs and Clientless VPNs." "Booleans in Compound Expressions.
VPN.CLIENTLESS_HOSTURL.HOSTNAME.PORT	"Expression Prefixes for VPNs and Clientless VPNs." "Compound Operations for Numbers."
VPN.CLIENTLESS_HOSTURL.PATH.IGNORE_EMPTY_ELEMENTS	Ignores spaces in the data. For an example,

Expression Prefix	Links to Relevant Information, with Applicable Notes and Operator Descriptions
VPN.CLIENTLESS_HOSTURL.QUERY.IGNORE_EMPTY_ELEMENTS	see the table "HTTP Expression Prefixes that Return Text." Ignores spaces in the data. For an example, see the table "HTTP Expression Prefixes that Return Text."
VPN.HOST	" Expression Prefixes for VPNs and Clientless VPNs."
VPN.HOST.[DOMAIN | Server]	"Expression Prefixes for VPNs and Clientless VPNs."
VPN.HOST.EQ("hostname")	"Expression Prefixes for VPNs and Clientless VPNs." "Booleans in Compound Expressions.
VPN.HOST.PORT	"Expression Prefixes for VPNs and Clientless VPNs." "Default Syntax Expressions: Evaluating Text." "Compound Operations for Numbers."

Expressions Reference-Classic Expressions

May 25, 2015

The subtopics listed in the table of contents on the left side of your screen contain tables listing the NetScaler classic expressions.

In the table of operators, the result type of each operator is shown at the beginning of the description. In the other tables, the level of each expression is shown at the beginning of the description. For named expressions, each expression is shown as a whole.

This document includes the following details:

Expression Element	Definition
==	Boolean. Returns TRUE if the current expression equals the argument. For text operations, the items being compared must exactly match one another. For numeric operations, the items must evaluate to the same number.
!=	Boolean. Returns TRUE if the current expression does not equal the argument. For text operations, the items being compared must not exactly match one another. For numeric operations, the items must not evaluate to the same number.
CONTAINS	Boolean. Returns TRUE if the current expression contains the string that is designated in the argument.
NOTCONTAINS	Boolean. Returns TRUE if the current expression does not contain the string that is designated in the

Expression Element	Definition
	argument.
CONTENTS	Text. Returns the contents of the current expression.
EXISTS	Boolean. Returns TRUE if the item designated by the current expression exists.
NOTEXISTS	Boolean. Returns TRUE if the item designated by the current expression does not exist.
>	Boolean. Returns TRUE if the current expression evaluates to a number that is greater than the argument.
<	Boolean. Returns TRUE if the current expression evaluates to a number that is less than the argument.
>=	Boolean. Returns TRUE if the current expression evaluates to a number that is greater than or equal to the argument.
<=	Boolean. Returns TRUE if the current expression evaluates to a number that is less than or equal to the argument.

Expression Element	Definition
REQ	Flow Type. Operates on incoming (or request) packets.
REQ.HTTP	Protocol Operates on HTTP requests.

Expression Element	Definition
REQ.HTTP.METHOD	Qualifier Designates the HTTP method.
REQ.HTTP.URL	Qualifier Designates the URL.
REQ.HTTP.URLTOKENS	Qualifier Designates the URL token.
REQ.HTTP.VERSION	Qualifier Designates the HTTP version.
REQ.HTTP.HEADER	Qualifier Designates the HTTP header.
REQ.HTTP.URLLEN	Qualifier Designates the number of characters in the URL.
REQ.HTTP.URLQUERY	Qualifier Designates the query portion of the URL.
REQ.HTTP.URLQUERYLEN	Qualifier Designates the length of the query portion of the URL.
REQ.SSL	Protocol Operates on SSL requests.
REQ.SSL.CLIENT.CERT	Qualifier Designates the entire client certificate.
REQ.SSL.CLIENT.CERT.SUBJECT	Qualifier Designates the client certificate subject.
REQ.SSL.CLIENT.CERT.ISSUER	Qualifier

Expression Element	Definition
	Designates the issuer of the client certificate.
REQ.SSL.CLIENT.CERT.SIGALGO	Qualifier Designates the validation algorithm used by the client certificate.
REQ.SSL.CLIENT.CERT.VERSION	Qualifier Designates the client certificate version.
REQ.SSL.CLIENT.CERT.VALIDFROM	Qualifier Designates the date before which the client certificate is not valid.
REQ.SSL.CLIENT.CERT.VALIDTO	Qualifier Designates the date after which the client certificate is not valid.
REQ.SSL.CLIENT.CERT.SERIALNUMBER	Qualifier Designates the serial number of the client certificate.
REQ.SSL.CLIENT.CIPHER.TYPE	Qualifier Designates the encryption protocol used by the client.
REQ.SSL.CLIENT.CIPHER.BITS	Qualifier Designates the number of bits used by the client's SSL key.
REQ.SSL.CLIENT.SSL.VERSION	Qualifier Designates the SSL version that the client is using.
REQ.TCP	Protocol Operates on incoming TCP packets.
REQ.TCP.SOURCEPORT	Qualifier Designates the source port of the incoming packet.
REQ.TCP.DESTPORT	Qualifier Designates the destination port of the incoming packet.

Expression Element	Definition
REQ.IP	Protocol Operates on incoming IP packets.
REQ.IP.SOURCEIP	Qualifier Designates the source IP of the incoming packet.
REQ.IP.DESTIP	Qualifier Designates the destination IP of the incoming packet.
RES	Flow Type Operates on outgoing (or response) packets.
RES.HTTP	Protocol Operates on HTTP responses.
RES.HTTP.VERSION	Qualifier Designates the HTTP version.
RES.HTTP.HEADER	Qualifier Designates the HTTP header.
RES.HTTP.STATUSCODE	Qualifier Designates the status code of the HTTP response.
RES.TCP	Protocol Operates on incoming TCP packets.
RES.TCP.SOURCEPORT	Qualifier Designates the source port of the outgoing packet.
RES.TCP.DESTPORT	Qualifier Designates the destination port of the outgoing packet.
RES.IP	Protocol

Expression Element	Definition
	Operates on outgoing IP packets.
RES.IP.SOURCEIP	Qualifier Designates the source IP of the outgoing packet. This can be in IPv4 or IPv6 format. For example: add expr exp3 “sourceip == 10.102.32.123 –netmask 255.255.255.0 && destip == 2001::23/120”.
RES.IP.DESTIP	Qualifier Designates the destination IP of the outgoing packet.

Updated: 2013-10-21

The expressions to configure client settings on the Access Gateway with the following software:

- Antivirus
- Personal firewall
- Antispam
- Internet Security

For example usage, see http://support.citrix.com/article/CTX112599.

Actual Expression	Definition
CLIENT.APPLICATION.AV(<NAME>.VERSION == <VERSION>)	Checks whether the client is running the designated anti-virus program and version.
CLIENT.APPLICATION.AV(<NAME>.VERSION != <VERSION>)	Checks whether the client is not running the designated anti-virus program and version.
CLIENT.APPLICATION.PF(<NAME>.VERSION == <VERSION>)	Checks whether the client is running the designated personal firewall program and version.
CLIENT.APPLICATION.PF(<NAME>.VERSION != <VERSION>)	Checks whether the client is not running the designated personal firewall program and version.
CLIENT.APPLICATION.IS(<NAME>.VERSION == <VERSION>)	Checks whether the client is running the designated internet security program and version.

Actual Expression	Definition
CLIENT.APPLICATION.IS(<NAME>.VERSION != <VERSION>)	Checks whether the client is not running the designated internet security program and version.
CLIENT.APPLICATION.AS(<NAME>.VERSION == <VERSION>)	Checks whether the client is running the designated anti-spam program and version.
CLIENT.APPLICATION.AS(<NAME>.VERSION != <VERSION>)	Checks whether the client is not running the designated anti-spam program and version.

Expression	Definition
REQ	Flow Type. Operates on incoming, or request, packets.
REQ.VLANID	Qualifier. Operates on the virtual LAN (VLAN) ID.
REQ.INTERFACE.ID	Qualifier. Operates on the ID of the designated NetScaler interface.
REQ.INTERFACE.RXTHROUGHPUT	Qualifier. Operates on the raw received packet throughput of the designated NetScaler interface.
REQ.INTERFACE.TXTHROUGHPUT	Qualifier. Operates on the raw transmitted packet throughput of the designated NetScaler interface.
REQ.INTERFACE.RXTXTHROUGHPUT	Qualifier. Operates on the raw received and transmitted packet throughput of the designated NetScaler interface.
REQ.ETHER.SOURCEMAC	Qualifier. Operates on the source MAC address.

Expression	Definition
REQ.ETHER.DESTMAC	Qualifier. Operates on the destination MAC address.
RES	Flow Type. Operates on outgoing (or response) packets.
RES.VLANID	Qualifier. Operates on the virtual LAN (VLAN) ID.
RES.INTERFACE.ID	Qualifier. Operates on the ID of the designated NetScaler interface.
RES.INTERFACE.RXTHROUGHPUT	Qualifier. Operates on the raw received packet throughput of the designated NetScaler interface.
RES.INTERFACE.TXTHROUGHPUT	Qualifier. Operates on the raw transmitted packet throughput of the designated NetScaler interface.
RES.INTERFACE.RXTXTHROUGHPUT	Qualifier. Operates on the raw received and transmitted packet throughput of the designated NetScaler interface.
RES.ETHER.SOURCEMAC	Qualifier. Operates on the source MAC address.
RES.ETHER.DESTMAC	Qualifier. Operates on the destination MAC address.

Expression	Definition
TIME	Qualifier.

Expression	Definition
	Operates on the date and time of day, GMT.
DATE	Qualifier. Operates on the date, GMT.
DAYOFWEEK	Operates on the specified day in the week, GMT.

Updated: 2013-09-30

You can specify file system expressions in authorization policies for users and groups who access file sharing through the NetScaler Gateway file transfer utility (the VPN portal). These expressions work with the NetScaler Gateway file transfer authorization feature to control user access to file servers, folders, and files. For example, you can use these expressions in authorization policies to control access based on file type and size.

Expression	Definition
FS.COMMAND	Qualifier. Operates on a file system command. The user can issue multiple commands on a file transfer portal. (For example, ls to list files or mkdir to create a directory). This expression returns the current action that the user is taking. Possible values: Neighbor, login, ls, get, put, rename, mkdir, rmdir, del, logout, any. Following is an example: `Add authorization policy pol1 "fs.command eq login && (fs.user eq administrator || fs.serverip eq 10.102.88.221 –netmask 255.255.255.252)" allow`
FS.USER	Returns the user who is logged on to the file system.
FS.SERVER	Returns the host name of the target server. In the following example, the string `win2k3-88-22` is the server name: `fs.server eq win2k3-88-221`
FS.SERVERIP	Returns the IP address of the target server.
FS.SERVICE	Returns a shared root directory on the file server. If a particular folder is exposed as shared, a user can directly log on to the specified first level folder. This first level folder is called a service. For example, in the path \\hostname\SERVICEX\ETC, SERVICEX is the service. As

Expression	Definition
	another example, if a user accesses the file \\hostname\service1\dir1\file1.doc, FS.SERVICE will return service1. Following is an example: fs.service notcontains New
FS.DOMAIN	Returns the domain name of the target server.
FS.PATH	Returns the complete path of the file being accessed. For example, if a user accesses the file \\hostname\service1\dir1\file1.doc, FS.PATH will return \service\dir1\file1.doc. Following is an example: fs.path notcontains SSL
FS.FILE	Returns the name of the file being accessed. For example, if a user accesses the file \\hostname\service1\dir1\file1.doc, FS.FILE will return file1.doc.
FS.DIR	Returns the directory being accessed. For example, if a user accesses the file \\hostname\service1\dir1\file1.doc, FS.DIR will return \service\dir1.
FS.FILE.ACCESSTIME	Returns the time at which the file was last accessed. This is one of several options that provide you with granular control over actions that the user performs. (See the following entries in this table.)
FS.FILE.CREATETIME	Returns the time at which the file was created.
FS.FILE.MODIFYTIME	Returns the time at which the file was edited.
FS.FILE.WRITETIME	Returns the time of the most recent change in the status of the file.
FS.FILE.SIZE	Returns the file size.
FS.DIR.ACCESSTIME	Returns the time at which the directory was last accessed.
FS.DIR.CREATETIME	Returns the time at which the directory was created.
FS.DIR.MODIFYTIME	Returns the time at which the directory was last modified.
FS.DIR.WRITETIME	Returns the time at which the directory status last changed.

Note: File system expressions do not support regular expressions.

Expression	Definition
ns_all_apps_ncomp	Tests for connections with destination ports between 0 and 65535. In other words, tests for all applications.
ns_cachecontrol_nocache	Tests for connections with an HTTP Cache-Control header that contains the value “no-cache”.
ns_cachecontrol_nostore	Tests for connections with an HTTP Cache-Control header that contains the value “no-store”.
ns_cmpclient	Tests the client to determine if it accepts compressed content.
ns_content_type	Tests for connections with an HTTP Content-Type header that contains “text”.
ns_css	Tests for connections with an HTTP Content-Type header that contains “text/css”.
ns_ext_asp	Tests for HTTP connections to any URL that contains the string .asp—in other words, any connection to an active server page (ASP).
ns_ext_cfm	Tests for HTTP connections to any URL that contains the string .cfm
ns_ext_cgi	Tests for HTTP connections to any URL that contains the string .cgi—in other words, any connection to a common gateway interface (CGI) script.
ns_ext_ex	Tests for HTTP connections to any URL that contains the string .ex
ns_ext_exe	Tests for HTTP connections to any URL that contains the string .exe—in other words, any connection to a executable file.
ns_ext_htx	Tests for HTTP connections to any URL that contains the string .htx
ns_ext_not_gif	Tests for HTTP connections to any URL that does not contain the string .gif—in other words, any connection to a URL that is not a GIF image.
ns_ext_not_jpeg	Tests for HTTP connections to any URL that does not contain the string .jpeg—in other words, any connection to a URL that is not a JPEG image.

Expression	Definition
ns_ext_shtml	Tests for HTTP connections to any URL that contains the string .shtml—in other words, any connection to a server-parsed HTML page.
ns_false	Always returns a value of FALSE.
ns_farclient	Client is in a different geographical region from the NetScaler, as determined by the geographical region in the client's IP address. The following regions are predefined: 192.0.0.0 – 193.255.255.255: Multi-regional 194.0.0.0 – 195.255.255.255: European Union 196.0.0.0 – 197.255.255.255: Other1 198.0.0.0 – 199.255.255.255: North America 200.0.0.0 – 201.255.255.255: Central and South America 202.0.0.0 – 203.255.255.255: Pacific Rim 204.0.0.0 – 205.255.255.255: Other2 206.0.0.0 – 207.255.255.255: Other3
ns_header_cookie	Tests for HTTP connections that contain a Cookie header
ns_header_pragma	Tests for HTTP connections that contain a Pragma: no-cache header.
ns_mozilla_47	Tests for HTTP connections whose User-Agent header contains the string Mozilla/4.7—in other words, any connection from a client using the Mozilla 4.7 Web browser.
ns_msexcel	Tests for HTTP connections whose Content-Type header contains the string application/vnd.msexcel—in other words, any connection transmitting a Microsoft Excel spreadsheet.
ns_msie	Tests for HTTP connections whose User-Agent header contains the string MSIE—in other words, any connection from a client using any version of the Internet Explorer Web browser.
ns_msppt	Tests for HTTP connections whose Content-Type header contains the string application/vnd.ms-powerpoint—in other words, any connection transmitting a Microsoft PowerPoint file.
ns_msword	Tests for HTTP connections whose Content-Type header contains the string

Expression	Definition
	application/vnd.msword—in other words, any connection transmitting a Microsoft Word file.
ns_non_get	Tests for HTTP connections that use any HTTP method except for GET.
ns_slowclient	Returns TRUE if the average round trip time between the client and the NetScaler is more than 80 milliseconds.
ns_true	Returns TRUE for all traffic.
ns_url_path_bin	Tests the URL path to see if it points to the /bin/ directory.
ns_url_path_cgibin	Tests the URL path to see if it points to the CGI-BIN directory.
ns_url_path_exec	Tests the URL path to see if it points to the /exec/ directory.
ns_url_tokens	Tests for the presence of URL tokens.
ns_xmldata	Tests for the presence of XML data.

Expression	Definition
McAfee Virus Scan 11	Tests to determine whether the client is running the latest version of McAfee VirusScan.
McAfee Antivirus	Tests to determine whether the client is running any version of McAfee Antivirus.
Symantec AntiVirus 10 (with Updated Definition File)	Tests to determine whether the client is running the most current version of Symantec AntiVirus.
Symantec AntiVirus 6.0	Tests to determine whether the client is running Symantec AntiVirus 6.0.
Symantec AntiVirus 7.5	Tests to determine whether the client is running Symantec AntiVirus 7.5.

Expression	Definition
TrendMicro OfficeScan 7.3	Tests to determine whether the client is running Trend Microsystems' OfficeScan, version 7.3.
TrendMicro AntiVirus 11.25	Tests to determine whether the client is running Trend Microsystems' AntiVirus, version 11.25.
Sophos Antivirus 4	Tests to determine whether the client is running Sophos Antivirus, version 4.
Sophos Antivirus 5	Tests to determine whether the client is running Sophos Antivirus, version 5.
Sophos Antivirus 6	Tests to determine whether the client is running Sophos Antivirus, version 6.

Expression	Definition
TrendMicro OfficeScan 7.3	Tests to determine whether the client is running Trend Microsystems' OfficeScan, version 7.3.
Sygate Personal Firewall 5.6	Tests to determine whether the client is running the Sygate Personal Firewall, version 5.6.
ZoneAlarm Personal Firewall 6.5	Tests to determine whether the client is running the ZoneAlarm Personal Firewall, version 6.5.

Expression	Definition
Norton Internet Security	Tests to determine whether the client is running any version of Norton Internet Security.

Summary Examples of Default Syntax Expressions and Policies

Jul 10, 2013

The following table provides examples of default syntax expressions that you can use as the basis for your own default syntax expressions.

Table 1. Examples of Default Syntax Expressions

Expression Type	Sample Expressions
Look at the method used in the HTTP request.	http.req.method.eq(post) http.req.method.eq(get)
Check the Cache-Control or Pragma header value in an HTTP request (req) or response (res).	http.req.header("Cache-Control").contains("no-store") http.req.header("Cache-Control").contains("no-cache") http.req.header("Pragma").contains("no-cache") http.res.header("Cache-Control").contains("private") http.res.header("Cache-Control").contains("public") http.res.header("Cache-Control").contains("must-revalidate") http.res.header("Cache-Control").contains("proxy-revalidate") http.res.header("Cache-Control").contains("max-age")
Check for the presence of a header in a request (req) or response (res).	http.req.header("myHeader").exists http.res.header("myHeader").exists
Look for a particular file type in an HTTP request based on the file extension.	http.req.url.contains(".html") http.req.url.contains(".cgi")

Expression Type	Sample Expressions
	http.req.url.contains(".asp") http.req.url.contains(".exe") http.req.url.contains(".cfm") http.req.url.contains(".ex") http.req.url.contains(".shtml") http.req.url.contains(".htx") http.req.url.contains("/cgi-bin/") http.req.url.contains("/exec/") http.req.url.contains("/bin/")
Look for anything that is other than a particular file type in an HTTP request.	http.req.url.contains(".gif").not http.req.url.contains(".jpeg").not
Check the type of file that is being sent in an HTTP response based on the Content-Type header.	http.res.header("Content-Type").contains("text") http.res.header("Content-Type").contains("application/msword") http.res.header("Content-Type").contains("vnd.ms-excel") http.res.header("Content-Type").contains("application/vnd.ms-powerpoint") http.res.header("Content-Type").contains("text/css") http.res.header("Content-Type").contains("text/xml") http.res.header("Content-Type").contains("image/")
Check whether this response contains an expiration header.	http.res.header("Expires").exists
Check for a Set-Cookie header in a response.	http.res.header("Set-Cookie").exists
Check the agent that sent the response.	http.res.header("User-

Expression Type	Sample Expressions
	Agent").contains("Mozilla/4.7") http.res.header("User-Agent").contains("MSIE")
Check if the first 1024 bytes of the body of a request starts with the string “some text”.	http.req.body(1024).contains("some text")

The following table shows examples of policy configurations and bindings for commonly used functions.

Table 2. Examples of Default Syntax Expressions and Policies

Purpose	Example
Use the rewrite feature to replace occurrences of http:// with https:// in the body of an HTTP response.	add rewrite action httpRewriteAction replace_all http.res.body(50000) "\"https://\"" -pattern http:// add rewrite policy demo_rep34312 "http.res.body(50000).contains(\"http://\")" httpRewriteAction
Replace all occurrences of “abcd” with “1234” in the first 1000 bytes of the HTTP body.	add rewrite action abcdTo1234Action replace_all "http.req.body(1000)" "\"1234\"" -pattern abcd add rewrite policy abcdTo1234Policy "http.req.body(1000).contains(\"abcd\")" abcdTo1234Action bind rewrite global abcdTo1234Policy 100 END -type REQ_OVERRIDE
Downgrade the HTTP version to 1.0 to prevent the server from chunking HTTP responses.	add rewrite action downgradeTo1.0Action replace http.req.version.minor "\"0\"" add rewrite policy downgradeTo1.0Policy "http.req.version.minor.eq(1)" downgradeTo1.0Action bind lb vserver myLBVserver -policyName downgradeTo1.0Policy -priority 100 -gotoPriorityExpression NEXT -type REQUEST
Remove references to the HTTP or HTTPS protocol in all responses, so that if the user's connection is HTTP, the link is opened by using HTTP, and if the user's connection is HTTPS, the link is opened by using HTTPS.	add rewrite action remove_http_https replace_all "http.res.body(1000000).set_text_mode(ignorecase)" "\"//\"" -pattern "re~https?://|HTTPS?://~" add rewrite policy remove_http_https true remove_http_https

Purpose	Example
	bind lb vserver test_vsvr -policyName remove_http_https -priority 20 -gotoPriorityExpression NEXT -type RESPONSE
Rewrite instances of http:// to https:// in all URLs. This policy uses the responder functionality.	add responder action httpToHttpsAction redirect "\"https://\" + http.req.hostname + http.req.url" -bypassSafetyCheck YES add responder policy httpToHttpsPolicy "!CLIENT.SSL.IS_SSL" httpToHttpsAction bind responder global httpToHttpsPolicy 1 END -type OVERRIDE
Modify a URL to redirect from URL A to URL B. In this example, “file5.html” is appended to the path. This policy uses the responder functionality.	add responder action appendFile5Action redirect "\"http://\" + http.req.hostname + http.req.url + \"/file5.html\"" -bypassSafetyCheck YES add responder policy appendFile5Policy "http.req.url.eq(\"/testsite\")" appendFile5Action bind responder global appendFile5Policy 1 END -type OVERRIDE
Redirect an external URL to an internal URL.	add rewrite action act_external_to_internal REPLACE 'http.req.hostname.server' '"www.my.host.com"' add rewrite policy pol_external_to_internal 'http.req.hostname.server.eq("www.external.host.com")' act_external_to_internal bind rewrite global pol_external_to_internal 100 END -type REQ_OVERRIDE
Redirect requests to www.example.com that have a query string to www.Webn.example.com. The value n is derived from a server parameter in the query string, for example, server=5.	add rewrite action act_redirect_query REPLACE q#http.req.header("Host").before_str(".example.com")' '"Web" + http.req.url.query.value("server")# add rewrite policy pol_redirect_query q#http.req.header("Host").eq("www.example.com") && http.req.url.contains("?")' act_redirect_query#
Limit the number of requests per second from a URL.	add ns limitSelector ip_limit_selector http.req.url "client.ip.src" add ns limitIdentifier ip_limit_identifier -threshold 4 -timeSlice 3600 -mode request_rate -limitType smooth -

Purpose	Example
	selectorName ip_limit_selector add responder action my_Web_site_redirect_action redirect "\"http://www.mycompany.com/\"" add responder policy ip_limit_responder_policy "http.req.url.contains(\"myasp.asp\") && sys.check_limit(\"ip_limit_identifier\")" my_Web_site_redirect_action bind responder global ip_limit_responder_policy 100 END -type default
Check the client IP address but pass the request without modifying the request.	add rewrite policy check_client_ip_policy 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS || HTTP.REQ.HEADER("client-ip").EXISTS' NOREWRITE bind rewrite global check_client_ip_policy 100 END
Remove old headers from a request and insert an NS-Client header.	add rewrite action del_x_forwarded_for delete_http_header x-forwarded-for add rewrite action del_client_ip delete_http_header client-ip add rewrite policy check_x_forwarded_for_policy 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS' del_x_forwarded_for add rewrite policy check_client_ip_policy 'HTTP.REQ.HEADER("client-ip").EXISTS' del_client_ip add rewrite action insert_ns_client_header insert_http_header NS-Client 'CLIENT.IP.SRC' add rewrite policy insert_ns_client_policy 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS || HTTP.REQ.HEADER("client-ip").EXISTS' insert_ns_client_header bind rewrite global check_x_forwarded_for_policy 100 200 bind rewrite global check_client_ip_policy 200 300 bind rewrite global insert_ns_client_policy 300 END
Remove old headers from a request, insert an NS-Client header, and then modify the “insert header” action so that the value of the inserted header	add rewrite action del_x_forwarded_for delete_http_header x-forwarded-for add rewrite action del_client_ip delete_http_header

Purpose	Example
contains the client IP values from the old headers and the NetScaler appliance's connection IP address. Note that this example repeats the previous example, with the exception of the final set rewrite action.	client_ip add rewrite policy check_x_forwarded_for_policy 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS' del_x_forwarded_for add rewrite policy check_client_ip_policy 'HTTP.REQ.HEADER("client-ip").EXISTS' del_client_ip add rewrite action insert_ns_client_header insert_http_header NS-Client 'CLIENT.IP.SRC' add rewrite policy insert_ns_client_policy 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS || HTTP.REQ.HEADER("client-ip").EXISTS' insert_ns_client_header bind rewrite global check_x_forwarded_for_policy 100 200 bind rewrite global check_client_ip_policy 200 300 bind rewrite global insert_ns_client_policy 300 END set rewrite action insert_ns_client_header -stringBuilderExpr 'HTTP.REQ.HEADER("x-forwarded-for").VALUE(0) + " " + HTTP.REQ.HEADER("client-ip").VALUE(0) + " " + CLIENT.IP.SRC' -bypassSafetyCheck YES

Tutorial Examples of Default Syntax Policies for Rewrite

May 26, 2015

With the rewrite feature, you can modify any part of an HTTP header, and, for responses, you can modify the HTTP body. You can use this feature to accomplish a number of useful tasks, such as removing unnecessary HTTP headers, masking internal URLs, redirecting Web pages, and redirecting queries or keywords.

In the following examples, you first create a rewrite action and a rewrite policy. Then you bind the policy globally.

This document includes the following details:

Updated: 2013-10-29

This example describes how to create a rewrite action and rewrite policy that redirects an external URL to an internal URL. You create an action, called act_external_to_internal, that performs the rewrite. Then you create a policy called pol_external_to_internal.

To redirect an external URL to an internal URL by using the command line interface

- To create the rewrite action, at the command prompt, type:

  add rewrite action act_external_to_internal REPLACE 'http.req.hostname.server' '"host_name_of_internal_Web_server"'

- To create the rewrite policy, at the NetScaler command prompt, type:

  add rewrite policy pol_external_to_internal 'http.req.hostname.server.eq("host_name_of_external_Web_server")' act_external_to_internal

- Bind the policy globally.

To redirect an external URL to an internal URL by using the configuration utility

1. Navigate to AppExpert > Rewrite > Actions.
2. In the details pane, click Add.
3. In the Create Rewrite Action dialog box, enter the name act_external_to_internal.
4. To replace the HTTP server host name with the internal server name, choose Replace from the Type list box.
5. In the Header Name field, type Host.
6. In the String expression for replacement text field, type the internal host name of your Web server.
7. Click Create and then click Close.

8. In the navigation pane, click Policies.
9. In the details pane, click Add.
10. In the Name field, type pol_external_to_internal. This policy will detect connections to the Web server.
11. In the Action drop-down menu, choose the action act_external_to_internal.
12. In the Expression editor, construct the following expression:

    HTTP.REQ.HOSTNAME.SERVER.EQ("www.example.com")
13. Bind your new policy globally.

This example describes how to create a rewrite action and rewrite policy that redirects a query to the proper URL. The example assumes that the request contains a Host header set to **www.example.com** and a GET method with the **string /query.cgi?server=5**. The redirect extracts the domain name from the host header and the number from the query string, and redirects the user's query to the server **Web5.example.com**, where the rest of the user's query is processed.

Note: Although the following commands appears on multiple lines, you should enter them on a single line without line breaks.

To redirect a query to the appropriate URL using the command line

- To create a rewrite action named act_redirect_query that replaces the HTTP server host name with the internal server name, type:

  add rewrite action act_redirect_query REPLACE q#http.req.header("Host").before_str(".example.com")' '"Web" + http.req.url.query.value("server")#

- To create a rewrite policy named pol_redirect_query, type the following commands at the NetScaler command prompt.. This policy detects connections, to the Web server, that contain a query string. Do not apply this policy to connections that do not contain a query string:

  add rewrite policy pol_redirect_query q#http.req.header("Host").eq("www.example.com") && http.req.url.contains("?")' act_redirect_query#

- Bind your new policy globally.

Because this rewrite policy is highly specific and should be run before any other rewrite policies, it is advisable to assign it a high priority. If you assign it a priority of 1, it will be evaluated first.

Updated: 2014-09-17

This example describes how to rewrite Web server responses to find all URLs that begin with the string “http” and replace that string with “https.” You can use this to avoid having to update Web pages after moving a server from HTTP to HTTPS.

To redirect HTTP URLs to HTTPS by using the command line interface

- To create a rewrite action named act_replace_http_with_https that replaces all instances of the string “http” with the string “https,” enter the following command:
  add rewrite action act_replace_http_with_https replace_all 'http.res.body(100)' '"https"' -pattern http

- To create a rewrite policy named pol_replace_http_with_https that detects connections to the Web server, enter the following command:
  add rewrite policy pol_replace_http_with_https TRUE act_replace_http_with_https NOREWRITE
- Bind your new policy globally.

To troubleshoot this rewrite operation, see "Case Study: Rewrite Policy for Converting HTTP Links to HTTPS not Working."

Updated: 2013-09-02

This example explains how to use a Rewrite policy to remove unwanted headers. Specifically, the example shows how to remove the following headers:

- **Accept Encoding header.** Removing the Accept Encoding header from HTTP responses prevents compression of the response.
- **Content Location header.** Removing the Content Location header from HTTP responses prevents your server from providing a hacker with information that might allow a security breach.

To delete headers from HTTP responses, you create a rewrite action and a rewrite policy, and you bind the policy globally.

To create the appropriate Rewrite action by using the command line interface

At the command prompt, type one of the following commands to either remove the Accept Encoding header and prevent response compression or remove the Content Location header:

- add rewrite action "act_remove-ae" delete_http_header "Accept-Encoding"
- add rewrite action "act_remove-cl" delete_http_header "Content-Location"

To create the appropriate Rewrite policy by using the command line interface

At the command prompt, type one of the following commands to remove either the Accept Encoding header or the Content Location header:

- add rewrite policy "pol_remove-ae" true "act_remove-ae"
- add rewrite policy "pol_remove-cl" true "act_remove-cl"

To bind the policy globally by using the command line interface

At the command prompt, type one of the following commands, as appropriate, to globally bind the policy that you have created:

- bind rewrite global pol_remove_ae 100
- bind rewrite global pol_remove_cl 200

This example explains how to use a Rewrite policy to modify connections to your home page and other URLs that end with a forward slash (/) to the default index page for your server, preventing redirects and reducing load on your server.

To modify directory-level HTTP requests to include the default home page by using

the command line

- To create a Rewrite action named action-default-homepage that modifies URLs that end in a forward slash to include the default home page index.html, type:

  add rewrite action "action-default-homepage" replace q#http.req.url.path "/" "/index.html"#

- To create a Rewrite policy named policy-default-homepage that detects connections to your home page and applies your new action, type:

  add rewrite policy "policy-default-homepage" q#http.req.url.path.EQ("/") "action-default-homepage"#

- Globally bind your new policy to put it into effect.

This example explains how to use a Rewrite policy to mask the information in the Server header in HTTP responses from your Web server. That header contains information that hackers can use to compromise your Web site. While masking the header will not prevent a skilled hacker from finding out information about your server, it will make hacking your Web server more difficult and encourage hackers to choose less well protected targets.

To mask the Server header in responses from the command line

1. To create a Rewrite action named act_mask-server that replaces the contents of the Server header with an uninformative string, type:

   add rewrite action "act_mask-server" replace "http.RES.HEADER(\"Server\")" "\"Web Server 1.0\""

2. To create a Rewrite policy named pol_mask-server that detects all connections, type:

   add rewrite policy "pol_mask-server" true "act_mask-server"

3. Globally bind your new policy to put it into effect.

Tutorial Examples of Classic Policies

May 26, 2015

The following examples describe useful examples of classic policy configuration for certain NetScaler features, such as NetScaler Gateway, application firewall, and SSL.

This document includes the following details:

- NetScaler Gateway Policy to Check for a Valid Client Certificate
- Application Firewall Policy to Protect a Shopping Cart Application
- Application Firewall Policy to Protect Scripted Web Pages
- DNS Policy to Drop Packets from Specific IPs
- SSL Policy to Require Valid Client Certificates

Updated: 2014-09-25

The following policies enable the NetScaler to ensure that a client presents a valid certificate before establishing a connection to a company's SSL VPN.

To check for a valid client certificate by using the command line interface

- Add an action to perform client certificate authentication.

  add ssl action act1 -clientAuth DOCLIENTAUTH

- Create an SSL policy to evaluate the client requests.

  add ssl policy pol1 -rule "REQ.HTTP.METHOD == GET" -action act1

- Add a rewrite action to insert the certificate issuer details into the HTTP header of the requests being sent to web server.

  add rewrite action act2 insert_http_header "CertDN" CLIENT.SSL.CLIENT_CERT.SUBJECT

- Create a rewrite policy to insert the certificate issuer details, if the client certificate exists.

  add rewrite policy pol2 "CLIENT.SSL.CLIENT_CERT.EXISTS" act2

  Bind these new policies to the NetScaler VIP to put them into effect.

Updated: 2013-09-02

Shopping cart applications handle sensitive customer information, for example, credit card numbers and expiration dates, and they access back-end database servers. Many shopping cart applications also use legacy CGI scripts, which can contain security flaws that were unknown at the time they were written, but are now known to hackers and identity thieves.

A shopping cart application is particularly vulnerable to the following attacks:

- **Cookie tampering.** If a shopping cart application uses cookies, and does not perform the appropriate checks on the

cookies that users return to the application, an attacker could modify a cookie and gain access to the shopping cart application under another user's credentials. Once logged on as that user, the attacker could obtain sensitive private information about the legitimate user or place orders using the legitimate user's account.

- **SQL injection.** A shopping cart application normally accesses a back-end database server. Unless the application performs the appropriate safety checks on the data users return in the form fields of its Web forms before it passes that information on to the SQL database, an attacker can use a Web form to inject unauthorized SQL commands into the database server. Attackers normally use this type of attack to obtain sensitive private information from the database or modify information in the database.

The following configuration will protect a shopping cart application against these and other attacks.

To protect a shopping cart application by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles, and then click Add.
2. In the Create Application Firewall Profile dialog box, in the Profile Name field, enter shopping_cart.
3. In the Profile Type drop-down list, select Web Application.
4. In the Configure Select Advanced defaults.
5. Click Create and then click Close.
6. In the details view, double-click the new profile.
7. In the Configure Web Application Profile dialog box, configure your new profile as described below:
   1. Click the Checks tab, double-click the Start URL check, and in the Modify Start URL Check dialog box, click the General tab and disable blocking, and enable learning, logging, statistics, and URL closure. Click OK and then click Close.

      Note that if you are using the command line, you configure these settings by typing the following at the prompt, and pressing ENTER:

      set appfw profile shopping_cart -startURLAction LEARN LOG STATS -startURLClosure ON

   2. For the Cookie Consistency check and Form Field Consistency checks, disable blocking, and enable learning, logging, statistics, using a similar method to the Modify Start URL Check configuration.

      If you are using the command line, you configure these settings by typing the following commands:

      set appfw profile shopping_cart -cookieConsistencyAction LEARN LOG STATS

      set appfw profile shopping_cart -fieldConsistencyAction LEARN LOG STATS

   3. For the SQL Injection check, disable blocking, and enable learning, logging, statistics, and transformation of special characters in the Modify SQL Injection Check dialog box, General tab, Check Actions section.

      If you are using the command line, you configure these settings by typing the following at the prompt, and pressing ENTER:

      set appfw profile shopping_cart -SQLInjectionAction LEARN LOG STATS -SQLInjectionTransformSpecialChars ON

   4. For the Credit Card check, disable blocking; enable logging, statistics, and masking of credit card numbers; and enable protection for those credit cards you accept as forms of payment.

      - If you are using the configuration utility, you configure blocking, logging, statistics, and masking (or x-out) in the Modify Credit Card Check dialog box, General tab, Check Actions section. You configure protection for specific credit cards in the Settings tab of the same dialog box.

- If you are using the command line, you configure these settings by typing the following at the prompt, and pressing ENTER:

  set appfw profile shopping_cart -creditCardAction LOG STATS -creditCardXOut ON -creditCard <name> [<name>...]

  For <name> you substitute the name of the credit card you want to protect. For Visa, you substitute VISA. For Master Card, you substitute MasterCard. For American Express, you substitute Amex. For Discover, you substitute Discover. For Diners Club, you substitute DinersClub. For JCB, you substitute JCB.

8. Create a policy named shopping_cart that detects connections to your shopping cart application and applies the shopping_cart profile to those connections.

   To detect connections to the shopping cart, you examine the URL of incoming connections. If you host your shopping cart application on a separate host (a wise measure for security and other reasons), you can simply look for the presence of that host in the URL. If you host your shopping cart in a directory on a host that handles other traffic, as well, you must determine that the connection is going to the appropriate directory and/or HTML page.

   The process for detecting either of these is the same; you create a policy based on the following expression, and substitute the proper host or URL for <string>.

   REQ.HTTP.HEADER URL CONTAINS <string>

   - If you are using the configuration utility, you navigate to the application firewall Policies page, click the Add... button to add a new policy, and follow the policy creation process described in “To create a policy with classic expressions using the configuration utility” beginning on page 201 and following.
   - If you are using the command line, you type the following command at the prompt and press Enter:
     add appfw policy shopping_cart "REQ.HTTP.HEADER URL CONTAINS <string>" shopping_cart

9. Globally bind your new policy to put it into effect.

Because you want to ensure that this policy will match all connections to the shopping cart, and not be preempted by another more general policy, you should assign a high priority to it. If you assign one (1) as the priority, no other policy can preempt this one.

Updated: 2013-11-14

Web pages with embedded scripts, especially legacy JavaScripts, often violate the “same origin rule,” which does not allow scripts to access or modify content on any server but the server where they are located. This security vulnerability is called cross-site scripting. The application firewall Cross-Site Scripting rule normally filters out requests that contain cross-site scripting.

Unfortunately, this can cause Web pages with older JavaScripts to stop functioning, even when your system administrator has checked those scripts and knows that they are safe. The example below explains how to configure the application firewall to allow cross-site scripting in Web pages from trusted sources without disabling this important filter for the rest of your Web sites.

To protect Web pages with cross-site scripting by using the command line interface

- At the command line, to create an advanced profile, type:

add appfw profile pr_xssokay -defaults advanced

- To configure the profile, type:

  set appfw profile pr_xssokay -startURLAction NONE -startURLClosure OFF -cookieConsistencyAction LEARN LOG STATS -fieldConsistencyAction LEARN LOG STATS -crossSiteScriptingAction LEARN LOG STATS$"

- Create a policy that detects connections to your scripted Web pages and applies the pr_xssokay profile, type:

  add appfw policy pol_xssokay "REQ.HTTP.HEADER URL CONTAINS ^\.pl\?$ || REQ.HTTP.HEADER URL CONTAINS ^\.js$" pr_xssokay

- Globally bind the policy.

To protect Web pages with cross-site scripting by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the details view, click Add.
3. In the Create Application Firewall Profile dialog box, create a Web Application profile with advanced defaults and name it pr_xssokay. Click Create and then click Close.
4. In the details view, click the profile, click Open, and in the Configure Web Application Profile dialog box, configure the pr_xssokay profile as shown below.

   Start URL Check: Clear all actions.

   - Cookie Consistency Check: Disable blocking.
   - Form Field Consistency Check: Disable blocking.
   - Cross-Site Scripting Check: Disable blocking.

   This should prevent blocking of legitimate requests involving Web pages with cross-site scripting that you know are nonetheless safe.

5. Click Policies, and then click Add.
6. In the Create Application Firewall Policy dialog box, create a policy that detects connections to your scripted Web pages and applies the pr_xssokay profile:

   - Policy name: pol_xssokay
   - Associated profile: pr_xssokay

   Policy expression: "REQ.HTTP.HEADER URL CONTAINS ^\.pl\?$ || REQ.HTTP.HEADER URL CONTAINS ^\.js$"

7. Globally bind your new policy to put it into effect.

Updated: 2013-09-02

The following example describes how to create a DNS action and DNS policy that detects connections from unwanted IPs or networks, such as those used in a DDOS attack, and drops all packets from those locations. The example shows networks within the IANA reserved IP block 192.168.0.0/16. A hostile network will normally be on publicly routable IPs.

To drop packets from specific IPs by using the command line interface

- To create a DNS policy named pol_ddos_drop that detects connections from hostile networks and drops those packets, type:

  add dns policy pol_ddos_drop 'client.ip.src.in_subnet(192.168.253.128/25) || client.ip.src.in_subnet(192.168.254.32/27)' -drop YES'

  For the example networks in the 192.168.0.0/16 range, you substitute the IP and netmask in ###.###.###.###/## format of each network you want to block. You can include as many networks as you want, separating each CLIENT.IP.SRC.IN_SUBNET(###.###.###.###./##) command with the OR operator.
- Globally bind your new policy to put it into effect.

Updated: 2013-09-02

The following example shows an SSL policy that checks the user's client certificate validity before initiating an SSL connection with a client.

To block connections from users with expired client certificates

- Log on to the command line interface.
  If you are using the GUI, navigate to the SSL Policies page, then in the Data area, click the Actions tab.
- Create an SSL action named act_current_client_cert that requires that users have a current client certificate to establish an SSL connection with the NetScaler.

  add ssl action act_current_client_cert-clientAuth DOCLIENTAUTH -clientCert ENABLED -certHeader "clientCertificateHeader" -clientCertNotBefore ENABLED -certNotBeforeHeader "Mon, 01 Jan 2007 00:00:00 GMT"
- Create an SSL policy named pol_current_client_cert that detects connections to the Web server that contain a query string.

  add ssl policy pol_current_ client_cert 'REQ.SSL.CLIENT.CERT.VALIDFROM >= "Mon, 01 Jan 2007 00:00:00 GMT"' act_block_ssl
- Bind your new policy globally.

  Because this SSL policy should apply to any user's SSL connection unless a more specific SSL policy applies, you may want to assign it a low priority. If you assign it a priority of one thousand (1000), that should ensure that other SSL policies are evaluated first, meaning that this policy will apply only to connections that do not match more specific policy criteria.

Migration of Apache mod_rewrite Rules to the Default Syntax

May 26, 2015

The Apache HTTP Server provides an engine known as mod_rewrite for rewriting HTTP request URLs. If you migrate the mod_rewrite rules from Apache to the NetScaler, you boost back-end server performance. In addition, because the NetScaler typically load balances multiple (sometimes thousands of) Web servers, after migrating the rules to the NetScaler you will have a single point of control for these rules.

Following are examples of mod_rewrite functions, and translations of these functions into Rewrite and Responder policies on the NetScaler.

This document includes the following details:

On some Web servers you can have multiple URLs for a resource. Although the canonical URLs should be used and distributed, other URLs can exist as shortcuts or internal URLs. You can make sure that users see the canonical URL regardless of the URL used to make an initial request.

In the following examples, the URL /~user is converted to /u/user.

Apache mod_rewrite solution for converting a URL

```
RewriteRule   ^/~([^/]+)/?(.*)    /u/$1/$2[R]
```

NetScaler solution for converting a URL

```
add responder action act1 redirect '"/u/"+HTTP.REQ.URL.AFTER_STR("/~")' -bypassSafetyCheck yes
add responder policy pol1 'HTTP.REQ.URL.STARTSWITH("/~") && HTTP.REQ.URL.LENGTH.GT(2)' act1
bind responder global pol1 100
```

You can enforce the use of a particular host name for reaching a site. For example, you can enforce the use of www.example.com instead of example.com.

Apache mod_rewrite solution for enforcing a particular host name for sites running on a port other than 80

```
RewriteCond %{HTTP_HOST}   !^www.example.com
RewriteCond %{HTTP_HOST}   !^$
RewriteCond %{SERVER_PORT} !^80$
RewriteRule ^/(.*)         http://www.example.com:%{SERVER_PORT}/$1 [L,R]
```

Apache mod_rewrite solution for enforcing a particular host name for sites running on port 80

```
RewriteCond %{HTTP_HOST}   !^www.example.com
RewriteCond %{HTTP_HOST}   !^$
RewriteRule ^/(.*)         http://www.example.com/$1 [L,R]
```

NetScaler solution for enforcing a particular host name for sites running on a port other than 80

```
add responder action act1 redirect '"http://www.example.com:"+CLIENT.TCP.DSTPORT+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol1 '!HTTP.REQ.HOSTNAME.CONTAINS("www.example.com")&&!HTTP.REQ.HOSTNAME.EQ("")&&!HTTP.REQ.HOSTNAME.PORT.EQ(80)&&HTTP.REQ.
bind responder global pol1 100 END
```

NetScaler solution for enforcing a particular host name for sites running on port 80

```
add responder action act1 redirect '"http://www.example.com"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol1  '!HTTP.REQ.HOSTNAME.CONTAINS("www.example.com")&&!HTTP.REQ.HOSTNAME.EQ("")&&HTTP.REQ.HOSTNAME.PORT.EQ(80)&&HTTP.REQ.
bind responder global  pol1 100 END
```

Usually the document root of a Web server is based on the URL "/". However, the document root can be any directory. You can redirect traffic to the document root if it changes from the top-level "/" directory to another directory.

In the following examples, you change the document root from / to /e/www. The first two examples simply replace one string with another. The third example is more universal because, along with replacing the root directory, it preserves the rest of the URL (the path and query string), for example, redirecting /example/file.html to /e/www/example/file.html.

Apache mod_rewrite solution for moving the document root

```
RewriteEngine on
RewriteRule   ^/$  /e/www/  [R]
```

NetScaler solution for moving the document root

```
add responder action act1 redirect '"/e/www/"' -bypassSafetyCheck yes
add responder policy pol1 'HTTP.REQ.URL.EQ("/")' act1
bind responder global pol1 100
```

NetScaler solution for moving the document root and appending path information to the request

```
add responder action act1 redirect '"/e/www"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol1 '!HTTP.REQ.URL.STARTSWITH("/e/www/")' act1
bind responder global pol1 100 END
```

You may want to redirect requests that are sent to home directories on a Web server to a different Web server. For example, if a new Web server is replacing an old one over time, as you migrate home directories to the new location you need to redirect requests for the migrated home directories to the new Web server.

In the following examples, the host name for the new Web server is newserver.

Apache mod_rewrite solution for redirecting to another Web server

```
RewriteRule   ^/(.+)  http://newserver/$1     [R,L]
```

NetScaler solution for redirecting to another Web server (method 1)

```
add responder  action act1 redirect '"http://newserver"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol1 'HTTP.REQ.URL.REGEX_MATCH(re#^/(.+)#)'   act1
bind responder global pol1 100 END
```

NetScaler solution for redirecting to another Web server (method 2)

```
add responder  action act1 redirect '"http://newserver"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol1 'HTTP.REQ.URL.LENGTH.GT(1)' act1
bind responder global pol1 100 END
```

Typically, a site with thousands of users has a structured home directory layout. For example, each home directory may reside under a subdirectory that is named using the first character of the user name. For example, the home directory for jsmith (/~jsmith/anypath) might be /home/j/smith/.www/anypath, and the home directory for rvalveti (/~rvalveti/anypath) might be /home/r/rvalveti/.www/anypath.

The following examples redirect requests to the home directory.

Apache mod_rewrite solution for structured home directories

```
RewriteRule   ^/~(([a-z])[a-z0-9]+)(.*)  /home/$2/$1/.www$3
```

NetScaler solution for structured home directories

NetScaler solution for structured home directories

```
add rewrite action act1 replace 'HTTP.REQ.URL'  '"/home/"+ HTTP.REQ.URL.AFTER_STR("~").PREFIX(1)+"/"+ HTTP.REQ.URL.AFTER_STR("~").BEFORE_STR("/")+"/.www"·
add rewrite policy pol1  'HTTP.REQ.URL.PATH.STARTSWITH("/~")' act1
bind rewrite global pol1 100
```

If a URL is not valid, it should be redirected to another Web server. For example, you should redirect to another Web server if a file that is named in a URL does not exist on the server that is named in the URL.

On Apache, you can perform this check using mod_rewrite. On the NetScaler, an HTTP callout can check for a file on a server by running a script on the server. In the following NetScaler examples, a script named file_check.cgi processes the URL and uses this information to check for the presence of the target file on the server. The script returns TRUE or FALSE, and the NetScaler uses the value that the script returns to validate the policy.

In addition to performing the redirection, the NetScaler can add custom headers or, as in the second NetScaler example, it can add text in the response body.

Apache mod_rewrite solution for redirection if a URL is wrong

```
RewriteCond   /your/docroot/%{REQUEST_FILENAME} !-f
RewriteRule   ^(.+)      http://webserverB.com/$1 [R]
```

NetScaler solution for redirection if a URL is wrong (method 1)

```
add HTTPCallout Call
set policy httpCallout Call -IPAddress 10.102.59.101 -port 80 -hostExpr '"10.102.59.101"' -returnType BOOL -ResultExpr 'HTTP.RES.BODY(100).CONTAINS("True")'  -urlSter
add responder action act1 redirect '"http://webserverB.com"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol1 '!HTTP.REQ.HEADER("Name").EXISTS  &&  !SYS.HTTP_CALLOUT(call)' act1
bind responder global pol1 100
```

NetScaler solution for redirection if a URL is wrong (method 2)

```
add HTTPCallout Call
set policy httpCallout Call -IPAddress 10.102.59.101 -port 80 -hostExpr '"10.102.59.101"' -returnType BOOL -ResultExpr 'HTTP.RES.BODY(100).CONTAINS("True")'  -urlSter
add responder  action act1 respondwith  '"HTTP/1.1 302 Moved Temporarily\r\nLocation: http://webserverB.com"+HTTP.REQ.URL+"\r\n\r\nHTTPCallout Used"' -bypassSafety
add responder policy pol1 '!HTTP.REQ.HEADER("Name").EXISTS  &&  !SYS.HTTP_CALLOUT(call)' act1
bind responder global pol1 100
```

You can rewrite a URL based on the time. The following examples change a request for example.html to example.day.html or example.night.html, depending on the time of day.

Apache mod_rewrite solution for rewriting a URL based on the time

```
RewriteCond   %{TIME_HOUR}%{TIME_MIN} >0700
RewriteCond   %{TIME_HOUR}%{TIME_MIN} <1900
RewriteRule   ^example\.html$ example.day.html [L]
RewriteRule   ^example\.html$ example.night.html
```

NetScaler solution for rewriting a URL based on the time

```
add rewrite action act1 insert_before 'HTTP.REQ.URL.PATH.SUFFIX(\'.\',0)' '"day."'
add rewrite action act2  insert_before 'HTTP.REQ.URL.PATH.SUFFIX(\'.\',0)' '"night."'
add rewrite  policy pol1 'SYS.TIME.WITHIN(LOCAL 07h 00m,LOCAL 18h 59m)' act1
add rewrite policy pol2 'true'  act2
bind rewrite global pol1 101
bind rewrite global pol2 102
```

If you rename a Web page, you can continue to support the old URL for backward compatibility while preventing users from recognizing that the page was renamed.

In the first two of the following examples, the base directory is /~quux/. The third example accommodates any base directory and the presence of query strings in the URL.

Apache mod_rewrite solution for managing a file name change in a fixed location

```
RewriteEngine  on
RewriteBase    /~quux/
RewriteRule    ^foo\.html$  bar.html
```

NetScaler solution for managing a file name change in a fixed location

```
add rewrite action act1 replace 'HTTP.REQ.URL.AFTER_STR("/~quux").SUBSTR("foo.html")' '"bar.html"'
add rewrite policy pol1 'HTTP.REQ.URL.ENDSWITH("/~quux/foo.html")' act1
bind rewrite global pol1 100
```

NetScaler solution for managing a file name change regardless of the base directory or query strings in the URL

```
add rewrite action act1 replace 'HTTP.REQ.URL.PATH.SUFFIX(\'/\',0)' '"bar.html"'
Add rewrite policy pol1 'HTTP.REQ.URL.PATH.CONTAINS("foo.html")' act1
Bind rewrite global pol1 100
```

If you rename a Web page, you may want to continue to support the old URL for backward compatibility and allow users to see that the page was renamed by changing the URL that is displayed in the browser.

In the first two of the following examples, redirection occurs when the base directory is /~quux/. The third example accommodates any base directory and the presence of query strings in the URL.

Apache mod_rewrite solution for changing the file name and the URL displayed in the browser

```
RewriteEngine on
RewriteBase    /~quux/
RewriteRule    ^old\.html$ new.html  [R]
```

NetScaler solution for changing the file name and the URL displayed in the browser

```
add responder action act1 redirect 'HTTP.REQ.URL.BEFORE_STR("foo.html")+"new.html"' -bypassSafetyCheck yes
add responder policy pol1 'HTTP.REQ.URL.ENDSWITH("/~quux/old.html")' act1
bind responder global pol1 100
```

NetScaler solution for changing the file name and the URL displayed in the browser regardless of the base directory or query strings in the URL

```
add responder action act1 redirect 'HTTP.REQ.URL.PATH.BEFORE_STR("old.html")+"new.html"+HTTP.REQ.URL.AFTER_STR("old.html")' -bypassSafetyCheck yes
add responder policy pol1 'HTTP.REQ.URL.PATH.CONTAINS("old.html")' act1
bind responder global pol1 100
```

To accommodate browser-specific limitations—at least for important top-level pages—it is sometimes necessary to set restrictions on the browser type and version. For example, you might want to set a maximum version for the latest Netscape variants, a minimum version for Lynx browsers, and an average feature version for all others.

The following examples act on the HTTP header "User-Agent", such that if this header begins with "Mozilla/3", the page MyPage.html is rewritten to MyPage.NS.html. If the browser is "Lynx" or "Mozilla" version 1 or 2, the URL becomes MyPage.20.html. All other browsers receive page MyPage.32.html.

Apache mod_rewrite solution for browser-specific settings

```
RewriteCond %{HTTP_USER_AGENT}  ^Mozilla/3.*
RewriteRule ^MyPage\.html$ MyPage.NS.html [L]
RewriteCond %{HTTP_USER_AGENT}  ^Lynx/.* [OR]
RewriteCond %{HTTP_USER_AGENT}  ^Mozilla/[12].*
RewriteRule ^MyPage\.html$ MyPage.20.html [L]
RewriteRule ^fMyPage\.html$ MyPage.32.html [L]
```

NetScaler solution for browser-specific settings

```
add patset pat1
bind patset pat1 Mozilla/1
bind Patset pat1 Mozilla/2
bind patset pat1 Lynx
bind Patset pat1 Mozilla/3
add rewrite action act1 insert_before 'HTTP.REQ.URL.SUFFIX' '"NS."'
add rewrite action act2 insert_before 'HTTP.REQ.URL.SUFFIX' '"20."'
add rewrite action act3 insert_before 'HTTP.REQ.URL.SUFFIX' '"32."'
add rewrite policy pol1 'HTTP.REQ.HEADER("User-Agent").STARTSWITH_INDEX("pat1").EQ(4)' act1
add rewrite policy pol2 'HTTP.REQ.HEADER("User-Agent").STARTSWITH_INDEX("pat1").BETWEEN(1,3)' act2
add rewrite policy pol3 '!HTTP.REQ.HEADER("User-Agent").STARTSWITH_ANY("pat1")' act3
bind rewrite global pol1 101 END
bind rewrite global pol2 102 END
bind rewrite global pol3 103 END
```

You can block a robot from retrieving pages from a specific directory or a set of directories to ease up the traffic to and from these directories. You can restrict access based on the specific location or you can block requests based on information in User-Agent HTTP headers.

In the following examples, the Web location to be blocked is /~quux/foo/arc/, the IP addresses to be blocked are 123.45.67.8 and 123.45.67.9, and the robot's name is NameOfBadRobot.

Apache mod_rewrite solution for blocking a path and a User-Agent header

```
RewriteCond %{HTTP_USER_AGENT}  ^NameOfBadRobot.*
RewriteCond %{REMOTE_ADDR}      ^123\.45\.67\.[8-9]$
RewriteRule ^/~quux/foo/arc/.+  -  [F]
```

NetScaler solution for blocking a path and a User-Agent header

```
add responder action act1 respondwith '"HTTP/1.1 403 Forbidden\r\n\r\n"'
add responder policy pol1 'HTTP.REQ.HEADER("User_Agent").STARTSWITH("NameOfBadRobot")&&CLIENT.IP.SRC.EQ(123.45.67.8)&&CLIENT.IP.SRC.EQ(123.45.67.9) && H
bind responder global pol1 100
```

If you find people frequently going to your server to copy inline graphics for their own use (and generating unnecessary traffic), you may want to restrict the browser's ability to send an HTTP Referer header.

In the following example, the graphics are located in http://www.quux-corp.de/~quux/.

Apache mod_rewrite solution for blocking access to an inline image

```
RewriteCond %{HTTP_REFERER} !^$
RewriteCond %{HTTP_REFERER} !^http://www.quux-corp.de/~quux/.*$
RewriteRule .*\.gif$ - [F]
```

NetScaler solution for blocking access to an inline image

```
add patset pat1
bind patset pat1 .gif
bind patset pat1 .jpeg
add responder action act1 respondwith '"HTTP/1.1 403 Forbidden\r\n\r\n"'
add responder policy pol1 '!HTTP.REQ.HEADER("Referer").EQ("") && !HTTP.REQ.HEADER("Referer").STARTSWITH("http://www.quux-corp.de/~quux/")&&HTTP.REQ.URL.EN
bind responder global pol1 100
```

To prevent users from knowing application or script details on the server side, you can hide file extensions from users. To do this, you may want to support extensionless links. You can achieve this behavior by using rewrite rules to add an extension to all requests, or to selectively add extensions to requests.

The first two of the following examples show adding an extension to all request URLs. In the last example, one of two file extensions is added. Note that in the last example, the mod_rewrite module can easily find the file extension because this module resides on the Web server. In contrast, the NetScaler must invoke an HTTP callout to check the extension of the requested file on the Web server. Based on the callout response, the NetScaler adds the .html or .php extension to the request URL.

Note: In the second NetScaler example, an HTTP callout is used to query a script named file_check.cgi hosted on the server. This script checks whether the argument that is provided in the callout is a valid file name.

Apache mod_rewrite solution for adding a .php extension to all requests

```
RewriteRule ^/?([a-z]+)$ $1.php [L]
```

NetScaler policy for adding a .php extension to all requests

```
add rewrite action act1 insert_after 'HTTP.REQ.URL' '".php"'
add rewrite policy pol1 'HTTP.REQ.URL.PATH.REGEX_MATCH(re#^/([a-z]+)$#)' act1
bind rewrite global pol1 100
```

Apache mod_rewrite solution for adding either .html or .php extensions to requests

```
RewriteCond %{REQUEST_FILENAME}.php -f
RewriteRule ^/?([a-zA-Z0-9]+)$ $1.php [L]
RewriteCond %{REQUEST_FILENAME}.html –f
RewriteRule ^/?([a-zA-Z0-9]+)$ $1.html [L]
```

NetScaler policy for adding either .html or .php extensions to requests

```
add HTTPCallout Call_html
add HTTPCallout Call_php
set policy httpCallout Call_html -IPAddress 10.102.59.101 -port 80 -hostExpr '"10.102.59.101"' -returnType BOOL -ResultExpr 'HTTP.RES.BODY(100).CONTAINS("True")'  -u
set policy httpCallout Call_php -IPAddress 10.102.59.101 -port 80 -hostExpr '"10.102.59.101"' -returnType BOOL -ResultExpr 'HTTP.RES.BODY(100).CONTAINS("True")'  -url
add patset pat1
bind patset pat1 .html
bind patset pat1 .php
bind patset pat1 .asp
bind patset pat1 .cgi
add rewrite  action act1 insert_after 'HTTP.REQ.URL.PATH'  '".html"'
add rewrite  action act2 insert_after "HTTP.REQ.URL.PATH"  '".php"'
add rewrite policy pol1 '!HTTP.REQ.URL.CONTAINS_ANY("pat1") && SYS.HTTP_CALLOUT(Call_html)' act1
add rewrite policy pol2 '!HTTP.REQ.URL.CONTAINS_ANY("pat1") && SYS.HTTP_CALLOUT(Call_php)' act2
bind rewrite global pol1 100 END
bind rewrite global pol2 101 END
```

Suppose that you have a set of working URLs that resemble the following:

/index.php?id=nnnn

To change these URLs to /nnnn and make sure that search engines update their indexes to the new URI format, you need to do the following:

- Redirect the old URIs to the new ones so that search engines update their indexes.
- Rewrite the new URI back to the old one so that the index.php script runs correctly.

To accomplish this, you can insert marker code into the query string (making sure that the marker code is not seen by visitors), and then removing the marker code for the index.php script.

The following examples redirect from an old link to a new format only if a marker is not present in the query string. The link that uses the new format is re-written back to the old format, and a marker is added to the query string.

Apache mod_rewrite solution

```
RewriteCond %{QUERY_STRING} !marker
RewriteCond %{QUERY_STRING} id=([-a-zA-Z0-9_+]+)
RewriteRule ^/?index\.php$ %1? [R,L]
RewriteRule ^/?([-a-zA-Z0-9_+]+)$  index.php?marker&id=$1 [L]
```

NetScaler solution

```
add responder action act_redirect redirect 'HTTP.REQ.URL.PATH.BEFORE_STR("index.php")+HTTP.REQ.URL.QUERY.VALUE("id")' -bypassSafetyCheck yes
add responder policy pol_redirect '!HTTP.REQ.URL.QUERY.CONTAINS("marker")&& HTTP.REQ.URL.QUERY.VALUE("id").REGEX_MATCH(re/[-a-zA-Z0-9_+]+/) && HTTP.REQ.U
bind responder global pol_redirect 100 END
add rewrite action act1 replace 'HTTP.REQ.URL.PATH.SUFFIX(\'/\',0)' '"index.phpmarker&id="+HTTP.REQ.URL.PATH.SUFFIX(\'/\',0)' -bypassSafetyCheck yes
add rewrite policy pol1 '!HTTP.REQ.URL.QUERY.CONTAINS("marker")'  act1
bind rewrite global pol1 100 END
```

To make sure that only secure servers are used for selected Web pages, you can use the following Apache mod_rewrite code or NetScaler Responder policies.

Apache mod_rewrite solution

```
RewriteCond %{SERVER_PORT} !^443$
RewriteRule ^/?(page1|page2|page3|page4|page5)$  https://www.example.com/%1 [R,L]
```

NetScaler solution using regular expressions

```
add responder action res_redirect redirect  '"https://www.example.com"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol_redirect '!CLIENT.TCP.DSTPORT.EQ(443)&&HTTP.REQ.URL.REGEX_MATCH(re/page[1-5]/)'  res_redirect
bind responder global pol_redirect 100 END
```

NetScaler solution using pattern sets

```
add patset pat1
bind patset pat1 page1
bind patset pat1 page2
bind patset pat1 page3
bind patset pat1 page4
bind patset pat1 page5
```

```
add responder action res_redirect redirect  '"https://www.example.com"+HTTP.REQ.URL' -bypassSafetyCheck yes
add responder policy pol_redirect '!CLIENT.TCP.DSTPORT.EQ(443)&&HTTP.REQ.URL.CONTAINS_ANY("pat1")'  res_redirect
bind responder global pol_redirect 100 END
```

Rate Limiting

Aug 12, 2014

The rate limiting feature enables you to define the maximum load for a given network entity or virtual entity on the Citrix NetScaler appliance. The feature enables you to configure the appliance to monitor the rate of traffic associated with the entity and take preventive action, in real time, based on the traffic rate. This feature is particularly useful when the network is under attack from a hostile client that is sending the appliance a flood of requests. You can mitigate the risks that affect the availability of resources to clients, and you can improve the reliability of the network and the resources that the appliance manages.

You can monitor and control the rate of traffic that is associated with virtual and user-defined entities, including virtual servers, URLs, domains, and combinations of URLs and domains. You can throttle the rate of traffic if it is too high, base information caching on the traffic rate, and redirect traffic to a given load balancing virtual server if the traffic rate exceeds a predefined limit. You can apply rate-based monitoring to HTTP, TCP, and DNS requests.

To monitor the rate of traffic for a given scenario, you configure a *rate limit identifier*. A rate limit identifier specifies numeric thresholds such as the maximum number of requests or connections (of a particular type) that are permitted in a specified time period called a *time slice*.

Optionally, you can configure filters, known as *stream selectors*, and associate them with rate limit identifiers when you configure the identifiers. After you configure the optional stream selector and the limit identifier, you must invoke the limit identifier from a default syntax policy. You can invoke identifiers from any feature in which the identifier may be useful, including rewrite, responder, DNS, and integrated caching.

You can globally enable and disable SNMP traps for rate limit identifiers. Each trap contains cumulative data for the rate limit identifier's configured data collection interval (time slice), unless you specified multiple traps to be generated per time slice. For more information about configuring SNMP traps and managers, see "SNMP."

Configuring a Stream Selector

Jul 30, 2014

A traffic stream selector is an optional filter for identifying an entity for which you want to throttle access. The selector is applied to a request or a response and selects data points (keys) that can be analyzed by a rate stream identifier. These data points can be based on almost any characteristic of the traffic, including IP addresses, subnets, domain names, TCP or UDP identifiers, and particular strings or extensions in URLs.

A stream selector consists of individual default syntax expressions called selectlets. Each selectlet is a non-compound default syntax expression. A traffic stream selector can contain up to five non-compound expressions called selectlets. Each selectlet is considered to be in an AND relationship with the other expressions. Following are some examples of selectlets:

```
http.req.url
http.res.body(1000>after_str(\"car_model\").before_str(\"made_in\")"
"client.ip.src.subnet(24)"
```

The order in which you specify parameters is significant. For example, if you configure an IP address and a domain (in that order) in one selector, and then specify the domain and the IP address (in the reverse order) in another selector, the NetScaler considers these values to be unique. This can lead to the same transaction being counted twice. Also, if multiple policies invoke the same selector, the NetScaler, again, can count the same transaction more than once.

Note: If you modify an expression in a stream selector, you may get an error if any policy that invokes it is bound to a new policy label or bind point. For example, suppose that you create a stream selector named myStreamSelector1, invoke it from myLimitID1, and invoke the identifier from a DNS policy named dnsRateLimit1. If you change the expression in myStreamSelector1, you might receive an error when binding dnsRateLimit1 to a new bind point. The workaround is to modify these expressions before creating the policies that invoke them.

At the command prompt, type:

add stream selector <name> <rule> ...

Example

```
> add stream selector myStreamSel HTTP.REQ.URL CLIENT.IP.SRC
```

Navigate to AppExpert > Rate Limiting > Selectors, click Add and specify the relevant details.

Configuring a Traffic Rate Limit Identifier

Sep 23, 2014

A rate limit identifier returns a Boolean TRUE if the amount of traffic exceeds a numeric limit within a particular time interval. The rate limit identifier definition can optionally include a stream selector. When you include a limit identifier in the compound default syntax expression in a policy rule, if you do not specify a stream selector, the limit identifier is applied to all the requests or responses that are identified by the compound expression.

Note: The maximum length for storing string results of selectors (for example, HTTP.REQ.URL) is 60 characters. If the string (for example, URL) is 1000 characters long, of which 50 characters are enough to uniquely identify a string, use an expression to extract only the required 50 characters.

At the command prompt, type:

add ns limitIdentifier <limitIdentifier> -threshold <positive_integer> -timeSlice <positive_integer> -mode <mode> -limitType (BURSTY | SMOOTH) -selectorName <string> -maxBandwidth <positive_integer> -trapsInTimeSlice <positive_integer>

Example

Configuring traffic rate limit identifier in BURSTY mode:

```
> add ns limitIdentifier 100_request_limit -threshold 100 -timeSlice 1000 -mode REQUEST_RATE -limitType BURSTY -selectorName limit_100_requests_selector -trapsInTime
```

Configuring traffic rate limit identifier in SMOOTH mode:

```
> add ns limitidentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200
```

Navigate to AppExpert > Rate Limiting > Limit Identifiers, click Add and specify the relevant details.

Configuring and Binding a Traffic Rate Policy

Aug 30, 2013

You implement rate-based application behavior by configuring a policy in an appropriate NetScaler feature. The feature must support default syntax policies. The policy expression must contain the following expression prefix to enable the feature to analyze the traffic rate:

```
sys.check_limit(<limit_identifier>)
```

Where `limit_identifier` is the name of a limit identifier.

The policy expression must be a compound expression that contains at least two components:

- An expression that identifies traffic to which the rate limit identifier is applied. For example: `http.req.url.contains("my_aspx.aspx")`.
- An expression that identifies a rate limit identifier, for example, `sys.check_limit("my_limit_identifier")`. This must be the last expression in the policy expression.

At the command prompt, type the following command to configure a rate-based policy and verify the configuration:

add cache | dns | rewrite | responder policy <policy_name> -rule expression && sys.check_limit("<LimitIdentifierName>") [<feature-specific information>]
Following is a complete example of a rate-based policy rule. Note that this example assumes that you have configured the responder action, send_direct_url, that is associated with the policy. Note that the sys.check_limit parameter must be the last element of the policy expression:

```
add responder policy responder_threshold_policy "http.req.url.contains(\"myindex.html\") && sys.check_limit(\"my_limit_identifier\")" send_direct_url
```

For information about binding a policy globally or to a virtual server, see "Binding Default Syntax Policies."

1. In the navigation pane, expand the feature in which you want to configure a policy (for example, Integrated Caching, Rewrite, or Responder), and then click Policies.
2. In the details pane, click Add. In Name, enter a unique name for the policy.
3. Under Expression, enter the policy rule, and make sure that you include the `sys.check_limit` parameter as the final component of the expression. For example:

   ```
   http.req.url.contains("my_aspx.aspx") && sys.check_limit("my_limit_identifier")
   ```
4. Enter feature-specific information about the policy.

   For example, you may be required to associate the policy with an action or a profile. For more information, see the feature-specific documentation.
5. Click Create, and then click Close.
6. Click Save.

Viewing the Traffic Rate

Aug 30, 2013

If traffic through one or more virtual servers matches a rate-based policy, you can view the rate of this traffic. The rate statistics are maintained in the limit identifier that you named in the rule for the rate-based policy. If more than one policy uses the same limit identifier, you can view the traffic rate as defined by hits to all of the policies that use the particular limit identifier.

At the command prompt, type the following command to view the traffic rate:

show ns limitSessions <limitIdentifier>

Example

```
sh limitsession myLimitSession
```

1. Navigate to AppExpert > Rate Limiting > Limit Identifiers.
2. Select a limit identifier whose traffic rate you want to view.
3. Click the Show Sessions button. If traffic through one or more virtual servers has matched a rate limiting policy that uses this limit identifier (and the hits are within the configured time slice for this identifier), the Session Details dialog box appears. Otherwise, you receive a "No session exists" message.

Testing a Rate-Based Policy

Oct 29, 2013

To test a rate-based policy, you can send traffic to any virtual server to which a rate-based policy is bound.

1. Configure a stream selector (optional) and a rate limit identifier (required). For example:

   ```
   add stream selector sel_subnet Q.URL "CLIENT.IP.SRC.SUBNET(24)"
   add ns limitIdentifier k_subnet -Threshold 4 -timeSlice 3600 -mode REQUEST_RATE  -limittype smooth  -selectorName sel_subnet  -trapsInTimeSlice 8
   ```
2. Configure the action that you want to associate with the policy that uses the rate limit identifier. For example:

   ```
   add responder action resp_redirect redirect "\"http://response_site.com/\""
   ```
3. Configure a policy that uses the sys.check_limit expression prefix to call the rate limit identifier. For example, the policy can apply a rate limit identifier to all requests arriving from a particular subnet, as follows:

   ```
   add responder policy resp_subnet "SYS.CHECK_LIMIT(\"k_subnet\")" resp_redirect
   ```
4. Bind the policy globally or to a virtual server. For example:

   ```
   bind responder global resp_subnet 6 END -type DEFAULT
   ```
5. In a browser address bar, send a test HTTP query to a virtual server. For example:

   ```
   http://<IP of a vserver>/testsite/test.txt
   ```
6. At the NetScaler command prompt, type:

   ```
   show ns limitSessions <limitIdentifier>
   ```

   Example

   ```
   > sh limitsession k_subnet
   1)     Time Remaining:      98 secs  Hits: 2                  Action Taken: 0
         Total Hash:    1718618  Hash String: /test.txt
         IPs gathered:
               1) 10.217.253.0
         Active Transactions: 0
    Done
   >
   ```
7. Repeat the query and check the limit identifier statistics again to verify that the statistics are being updated correctly.

Examples of Rate-Based Policies

Apr 18, 2013

The following table shows examples of rate-based policies.

Table 1. Examples of Rate-Based Policies

Purpose	Example
Limit the number of requests per second from a URL	add stream selector ipStreamSelector http.req.url "client.ip.src" add ns limitIdentifier ipLimitIdentifier -threshold 4 -timeSlice 1000 -mode request_rate -limitType smooth -selectorName ipStreamSelector add responder action myWebSiteRedirectAction redirect "\"http://www.mycompany.com/\"" add responder policy ipLimitResponderPolicy "http.req.url.contains(\"myasp.asp\") && sys.check_limit(\"ipLimitIdentifier\")" myWebSiteRedirectAction bind responder global ipLimitResponderPolicy 100 END -type default
Cache a response if the request URL rate exceeds 5 per 20000 milliseconds	add stream selector cacheStreamSelector http.req.url add ns limitidentifier cacheRateLimitIdentifier -threshold 5 -timeSlice 2000 -selectorName cacheStreamSelector add cache policy cacheRateLimitPolicy -rule "http.req.method.eq(get) && sys.check_limit(\"cacheRateLimitIdentifier\")" -action cache bind cache global cacheRateLimitPolicy -priority 10
Drop a connection on the basis of cookies received in requests from www.yourcompany.com if the requests exceed the rate limit	add stream selector reqCookieStreamSelector "http.req.cookie .value(\"mycookie\")" "client.ip.src.subnet(24)" add ns limitIdentifier myLimitIdentifier -Threshold 2 -timeSlice 3000 -selectorName reqCookieStreamSelector add responder action sendRedirectUrl redirect '\"http://www.mycompany.com\" + http.req.url' -bypassSafetyCheck YES add responder policy rateLimitCookiePolicy "http.req.url.contains(\"www.yourcompany.com\") && sys.check_limit(\"myLimitIdentifier\")" sendRedirectUrl
Drop a DNS packet if the requests from a particular client IP address and DNS domain exceed the rate limit	add stream selector dropDNSStreamSelector client.udp.dns.domain client.ip.src add ns limitIdentifier dropDNSRateIdentifier -timeslice 20000 -mode request_rate -selectorName dropDNSStreamSelector -maxBandwidth 1 -trapsintimeslice 20 add dns policy dnsDropOnClientRatePolicy "sys.check_limit (\"dropDNSRateIdentifier\")" -drop yes

limit	
Purpose	Example
Limit the number of HTTP requests that arrive from the same subnet (with a subnet mask of 32) and that have the same destination IP address.	add stream selector ipv6_sel "CLIENT.IPv6.src.subnet(32)" CLIENT.IPv6.dst Q.URL add ns limitIdentifier ipv6_id -imeSlice 20000 -selectorName ipv6_sel add lb vserver ipv6_vip HTTP 3ffe::209 80 -persistenceType NONE -cltTimeout 180 add responder action redirect_page redirect "\"http://redirectpage.com/\"" add responder policy ipv6_resp_pol "SYS.CHECK_LIMIT(\"ipv6_id\")" redirect_page bind responder global ipv6_resp_pol 5 END -type DEFAULT

Sample Use Cases for Rate-Based Policies

Dec 10, 2013

The following scenarios describe two uses of rate-based policies in global server load balancing (GSLB):

- The first scenario describes the use of a rate-based policy that sends traffic to a new data center if the rate of DNS requests exceed 1000 per second.
- In the second scenario, if more than five DNS requests arrive for a local DNS (LDNS) client within a particular period, the additional requests are dropped.

In this scenario, you configure a proximity-based load balancing method, and a rate-limiting policy that identifies DNS requests for a particular region. In the rate-limiting policy, you specify a threshold of 1000 DNS requests per second. A DNS policy applies the rate limiting policy to DNS requests for the region "Europe.GB.17.London.UK-East.ISP-UK." In the DNS policy, DNS requests that exceed the rate limiting threshold, starting with request 1001 and continuing to the end of the one-second interval, are to be forwarded to the IP addresses that are associated with the region "North America.US.TX.Dallas.US-East.ISP-US."

The following configuration demonstrates this scenario:

```
add stream selector DNSSelector1 client.udp.dns.domain
add ns limitIdentifier DNSLimitIdentifier1 -threshold 5 -timeSlice 1000 -selectorName DNSSelector1
add dns policy DNSLimitPolicy1 "client.ip.src.matches_location(\"Europe.GB.17.London.*.*\") &&
sys.check_limit(\"DNSLimitIdentifier1\")" -preferredLocation "North America.US.TX.Dallas.*.*"
bind dns global DNSLimitPolicy1 5
```

In the following example of global server load balancing, you configure a rate limiting policy that permits a maximum of five DNS requests in a particular interval, per domain, to be directed to an LDNS client for resolution. Any requests that exceed this rate are dropped. This type of policy can help protect the NetScaler from resource exploitation. For example, in this scenario, if the time to live (TTL) for a connection is five seconds, this policy prevents the LDNS from requerying a domain. Instead, it uses data that is cached on the NetScaler.

```
add stream selector LDNSSelector1 client.udp.dns.domain client.ip.src
add ns limitIdentifier LDNSLimitIdentifier1 -threshold 5 -timeSlice 1000 -selectorName LDNSSelector1
add dns policy LDNSPolicy1 "client.udp.dns.domain.contains(\".\") && sys.check_limit(\"LDNSLimitIdentifier1\")" -drop YES
bind dns global LDNSPolicy1 6
show gslb vserver gvip
gvip - HTTP     State: UP
Last state change was at Mon Sep  8 11:50:48 2008 (+711 ms)
Time since last state change: 1 days, 02:55:08.830
Configured Method: STATICPROXIMITY
BackupMethod: ROUNDROBIN
No. of Bound Services :  3 (Total)      3 (Active)
Persistence: NONE       Persistence ID: 100
Disable Primary Vserver on Down: DISABLED       Site Persistence: NONE
Backup Session Timeout: 0
Empty Down Response: DISABLED
Multi IP Response: DISABLED Dynamic Weights: DISABLED
Cname Flag: DISABLED
Effective State Considered: NONE
1)     site11_svc(10.100.00.00: 80)- HTTP State: UP    Weight: 1
Dynamic Weight: 0       Cumulative Weight: 1
Effective State: UP
Threshold : BELOW
Location: Europe.GB.17.London.UK-East.ISP-UK
2)     site12_svc(10.101.00.100: 80)- HTTP State: UP    Weight: 1
Dynamic Weight: 0       Cumulative Weight: 1
```

Effective State: UP
Threshold : BELOW
Location: North America.US.TX.Dallas.US-East.ISP-US
3) site13_svc(10.102.00.200: 80)- HTTP State: UP Weight: 1
Dynamic Weight: 0 Cumulative Weight: 1
Effective State: UP
Threshold : BELOW
Location: North America.US.NJ.Salem.US-Mid.ISP-US
1) www.gslbindia.com TTL: 5 secn
Cookie Timeout: 0 min Site domain TTL: 3600 sec
Done

Responder

Mar 20, 2012

Today's complex Web configurations often require different responses to HTTP requests that appear, on the surface, to be similar. When users request a Web site's home page, you may want to provide a different home page depending on where each user is located, which browser the user is using, or which language(s) the browser accepts and the order of preference. You might want to break the connection immediately if the request is coming from an IP range that has been generating DDoS attacks or initiating hacking attempts.

With the Responder feature, responses can be based on who sends the request, where it is sent from, and other criteria with security and system management implications. The feature is simple and quick to use. By avoiding the invocation of more complex features, it reduces CPU cycles and time spent in handling requests that do not require complex processing.

For handling sensitive data such as financial information, if you want to ensure that the client uses a secure connection to browse a site, you can redirect the request to secure connection by using https:// instead of http://.

To use the Responder feature, do the following;

- Enable the Responder feature on the NetScaler.
- Configure responder actions. The action can be to generate a custom response, redirect a request to a different Web page, or reset a connection.
- Configure responder policies. The policy determines the requests (traffic) on which an action has to be taken.
- Bind each policy to a bind point put it into effect. A bind point refers to an entity at which NetScaler examines the traffic to see if it matches a policy. For example, a bind point can be a load balancing virtual server.

You can specify a default action for requests that do not match any policy, and you can bypass the safety check for actions that would otherwise generate error messages.

The Rewrite feature of NetScaler helps in rewriting some information in the requests or responses handled by NetScaler. The following section shows some differences between the two features.

The main difference between the rewrite feature and the responder feature is as follows:

Responder cannot be used for response or server-based expressions. Responder can be used only for the following scenarios depending on client parameters:

- Redirecting a http request to new Web sites or Web pages
- Responding with some custom response
- Dropping or resetting a connection at request level

In case of a responder policy, the NetScaler examines the request from the client, takes action according to the applicable policies, sends the response to the client, and closes the connection with the client.

In case of a rewrite policy, the NetScaler examines the request from the client or response from the server, takes action according to the applicable policies, and forwards the traffic to the client or the server.

In general, it is recommended to use responder if you want the NetScaler to reset or drop a connection based on a client or request-based parameter. Use responder to redirect traffic, or respond with custom messages. Use rewrite for manipulating data on HTTP requests and responses.

Enabling the Responder Feature

Oct 29, 2013

To use the Responder feature, you must first enable it.

At the command prompt, type the following commands to enable the responder feature and verify the configuration:

- enable ns feature<feature>
- show ns feature

Example

```
enable ns feature Responder
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection          SP              ON
 .
 .
 .
 22)    Responder                RESPONDER          ON
 23)   HTML Injection            HTMLInjection      ON
 24)   NetScaler Push            push               OFF
 Done
>
```

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change advanced features.
3. In the Configure Advanced Features dialog box, select the Responder check box, and then click OK.
4. In the Enable/Disable Feature(s)? dialog box, click YES. A message appears in the status bar, stating that the feature has been enabled.

Configuring a Responder Action

Sep 08, 2016

After enabling the responder feature, you must configure one or more actions for handling requests. The responder supports the following types of actions:

Respond with

Sends the response defined by the Target expression without forwarding the request to a web server. (The NetScaler appliance substitutes for and acts as a web server.) Use this type of action to manually define a simple HTML-based response. Normally the text for a Respond with action consists of a web server error code and brief HTML page.

Respond with SQL OK

Sends the designated SQL OK response defined by the Target expression. Use this type of action to send an SQL OK response to an SQL query.

Respond with SQL Error

Sends the designated SQL Error response defined by the Target expression. Use this type of action to send an SQL Error response to an SQL query.

Respond with HTML page

Sends the designated HTML page as the response. You can choose from a drop-down list of HTML pages that were previously uploaded, or upload a new HTML page. Use this type of action to send an imported HTML page as the response.

Redirect

Redirects the request to a different web page or web server. A Redirect action can redirect requests originally sent to a "dummy" web site that exists in DNS, but for which there is no actual web server, to an actual web site. It can also redirect search requests to an appropriate URL. Normally, the redirection target for a Redirect action consists of a complete URL.

At the command prompt, type the following commands to configure a responder action and verify the configuration:

- add responder action <name> <type> <target> [-bypassSafetyCheck (YES | NO)]
- show responder action

Example

To create a responder action that displays a “Not Found” error page for URLs that do not exist:

```
add responder action act404Error respondWith

'"HTTP/1.1 404 Not Found\r\n\r\n"+ "HTTP.REQ.URL.HTTP_URL_SAFE"

+ "does not exist on the web server."'

 Done
```

```
> show responder action



1)     Name: act404Error

       Operation: respondwith

       Target: "HTTP/1.1 404 Not Found



"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."

       BypassSafetyCheck : NO

       Hits: 0

       Undef Hits: 0

       Action Reference Count: 0

 Done
```

To create a responder action that displays a “Not Found” error page for URLs that do not exist:

```
add responder action act404Error respondWith

'"HTTP/1.1 404 Not Found\r\n\r\n"+ "HTTP.REQ.URL.HTTP_URL_SAFE" +

 "does not exist on the web server."'

 Done

> show responder action
```

```
1)     Name: act404Error

       Operation: respondwith

       Target: "HTTP/1.1 404 Not Found



"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."

       BypassSafetyCheck : NO

       Hits: 0

       Undef Hits: 0

       Action Reference Count: 0

 Done

Configuring using Command Line Interface

set ns tcpprofile nstcp_default_profile -burstRateControl Dynamic -tcprate 0 -rateqmax 0
```

```
add responder action act404Error respondWith '"HTTP/1.1 404 Not Found\r\n\r\n"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does n
 Done
> show responder action


1)     Name: act404Error
       Operation: respondwith
       Target: "HTTP/1.1 404 Not Found


"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."
       BypassSafetyCheck : NO
       Hits: 0
       Undef Hits: 0
       Action Reference Count: 0
 Done
```

At the command prompt, type the following command to modify an existing responder action and verify the configuration:

- set responder action <name> -target <string> [-bypassSafetyCheck (YES | NO)]
- show responder action

Example

```
set responder action act404Error -target '"HTTP/1.1 404 Not Found\r\n\r\n"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not ex

Done

> show responder action



1) Name: act404Error

Operation: respondwith

Target: "HTTP/1.1 404 Not Found



"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."

BypassSafetyCheck : NO

Hits: 0

Undef Hits: 0

Action Reference Count: 0

Done
```

At the command prompt, type the following command to remove a responder action and verify the configuration:

- rm responder action <name>
- show responder action

Example

1. Navigate to AppExpert > Responder > Actions.
2. In the details pane, do one of the following:
   - To create a new action, click Add.
   - To modify an existing action, select the action, and then click Open.
3. Click Create or OK, depending on whether you are creating a new action or modifying an existing action.
4. Click Close. A message appears in the status bar, stating that the feature has been enabled.
5. To delete a responder action, select the action, and then click Remove. A message appears in the status bar, stating that the feature has been disabled.

1. In the Create Responder Action or Configure Responder Action dialog box, click Add.
2. In the Add Expression dialog box, in the first list box choose the first term for your expression.

   HTTP

   The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.

   SYS

   The protected web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.

   CLIENT

   The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.

   ANALYTICS

   The analytics data associated with the request. Choose this if you want to examine request metadata.

   SIP

   A SIP request. Choose this if you want to examine some aspect of a SIP request.

   When you make your choice, the rightmost list box lists appropriate terms for the next part of your expression.
3. In the second list box, choose the second term for your expression. The choices depend upon which choice you made in the previous step, and are appropriate to the context. After you make your second choice, the Help window below the Construct Expression window (which was blank) displays help describing the purpose and use of the term you just chose.
4. Continue choosing terms from the list boxes that appear to the right of the previous list box, or typing strings or numbers in the text boxes that appear to prompt you to enter a value, until your expression is finished.

You can configure the global HTTP action to invoke a responder action when an HTTP request times out. To configure this feature, you must first create the responder action that you want to invoke. Then, you configure the global HTTP timeout action to respond to a timeout with that responder action.

To configure the global HTTP action by using the command line interface

At the command prompt, type the following command:

- set ns httpProfile -reqTimeoutAction <responder action name>
- save ns config

For **<responder action name>**, substitute the name of the responder action.

Configuring a Responder Policy

Oct 29, 2013

After you configure a responder action, you must next configure a responder policy to select the requests to which the NetScaler appliance should respond. A responder policy is based on a rule, which consists of one or more expressions. The rule is associated with an action, which is performed if a request matches the rule.

Note: For creating and managing responder policies, the configuration utility provides assistance that is not available at the NetScaler command prompt.

At the command prompt, type the following command to add a new responder policy and verify the configuration:

- add responder policy <name> <expression> <action> [<undefaction>]-appFlowaction<actionName>
- show responder policy <name>

Example

```
> add responder policy policyThree "CLIENT.IP.SRC.IN_SUBNET(222.222.0.0/16)" RESET
 Done
> show responder policy policyThree

      Name: policyThree
      Rule: CLIENT.IP.SRC.IN_SUBNET(222.222.0.0/16)
      Responder Action: RESET
      UndefAction: Use Global
      Hits: 0
      Undef Hits: 0
 Done
```

At the command prompt, type the following command to modify an existing responder policy and verify the configuration:

- set responder policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>]
- show responder policy <name>

At the command prompt, type the following command to remove a responder policy and verify the configuration:

- rm responder policy <name>
- show responder policy

Example

```
>rm responder policy pol404Error
 Done

> show responder policy
 Done
```

1. Navigate to AppExpert > Responder > Policies.
2. In the details pane, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the policy, and then click Open.
3. Click Create or OK, depending on whether you are creating a new policy or modifying an existing policy.
4. Click Close. A message appears in the status bar, stating that the feature has been configured.

Binding a Responder Policy

Oct 29, 2013

To put a policy into effect, you must bind it either globally, so that it applies to all traffic that flows through the NetScaler, or to a specific virtual server, so that the policy applies only to requests whose destination IP address is the VIP of that virtual server.

When you bind a policy, you assign a priority to it. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer.

In the NetScaler operating system, policy priorities work in reverse order—the higher the number, the lower the priority. For example, if you have three policies with priorities of 10, 100, and 1000, the policy assigned a priority of 10 is performed first, then the policy assigned a priority of 100, and finally the policy assigned an order of 1000. The responder feature implements only the first policy that a request matches, not any additional policies that it might also match, so policy priority is important for getting the results you intend.

You can leave yourself plenty of room to add other policies in any order, and still set them to evaluate in the order you want, by setting priorities with intervals of 50 or 100 between each policy when you globally bind it. You can then add additional policies at any time without having to reassign the priority of an existing policy.

For additional information about binding policies on the NetScaler, see "Policies and Expressions."
Note: Responder policies cannot be bound to TCP-based virtual servers.

At the command prompt, type the following command to globally bind a responder policy and verify the configuration:

- bind responder global <policyName> <priority> [<gotoPriorityExpression [-type <type>] [-invoke (<labelType> <labelName>)]
- show responder global

Example

```
> bind responder global poliError 100
 Done
> show responder global
1)     Global bindpoint: REQ_DEFAULT
       Number of bound policies: 1

 Done
```

At the command prompt, type the following command to bind responder policy to a specific virtual server and verify the configuration:
bind lb vserver <name> -policyname <policy_name> -priority <priority>
Example

```
> bind lb vserver vs-loadbal -policyName policyTwo -priority 100
 Done
> show lb vserver
```

```
1)    vs-loadbal (10.102.29.20:80) - HTTP    Type: ADDRESS
      State: OUT OF SERVICE
      Last state change was at Wed Aug 19 09:05:47 2009 (+211 ms)
      Time since last state change: 2 days, 00:58:03.260
      Effective State: DOWN
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Port Rewrite : DISABLED
      No. of Bound Services :  0 (Total)      0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
      Vserver IP and Port insertion: OFF
      Push: DISABLED  Push VServer:
      Push Multi Clients: NO
      Push Label Rule: none
2)    vs-cont-sw (0.0.0.0:0) - TCP    Type: ADDRESS
      State: DOWN
      Last state change was at Wed Aug 19 10:03:46 2009 (+213 ms)
      Time since last state change: 2 days, 00:00:04.260
      Effective State: DOWN
      Client Idle Timeout: 9000 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      No. of Bound Services :  0 (Total)      0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
      Connection Failover: DISABLED
 Done
```

1. Navigate to AppExpert > Responder > Policies.
2. On the Responder Policies page, select a responder policy, and then click Policy Manager.
3. In the Responder Policy Manager dialog box Bind Points menu, select Default Global.
4. Click Insert Policy to insert a new row and display a drop-down list of all unbound responder policies.
5. Click one of the policies on the list. That policy is inserted into the list of globally bound responder policies.
6. Click Apply Changes.
7. Click Close. A message appears in the status bar, stating that the configuration has been successfully completed.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. On the Load Balancing Virtual Servers page, select the virtual server to which you want to bind the responder policy, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, select the Policies tab, which displays a list of all policies configured on your NetScaler appliance.
4. Select the check box next to the name of the policy you want to bind to this virtual server.

5. Click OK. A message appears in the status bar, stating that the configuration has been successfully completed.

Setting the Default Action for a Responder Policy

Oct 25, 2016

The NetScaler appliance generates an undefined event (UNDEF event) when a request does not match a responder policy, and then carries out the default action assigned to undefined events. By default, that action is to forward the request to the next feature without changing it. This default behavior is normally what you want; it ensures that requests that do not require special handling by a specific responder action are sent to your Web servers and clients receive access to the content that they requested.

If the Web site(s) your NetScaler appliance protects receive a significant number of invalid or malicious requests, however, you may want to change the default action to either reset the client connection or drop the request. In this type of configuration, you would write one or more responder policies that would match any legitimate requests, and simply redirect those requests to their original destinations. Your NetScaler appliance would then block any other requests as specified by the default action you configured.

You can assign any one of the following actions to an undefined event:

NOOP

The NOOP action aborts responder processing but does not alter the packet flow. This means that the appliance continues to process requests that do not match any responder policy, and eventually forwards them to the requested URL unless another feature intervenes and blocks or redirects the request. This action is appropriate for normal requests to your Web servers and is the default setting.

RESET

If the undefined action is set to RESET, the appliance resets the client connection, informing the client that it must re-establish its session with the Web server. This action is appropriate for repeat requests for Web pages that do not exist, or for connections that might be attempts to hack or probe your protected Web site(s).

DROP

If the undefined action is set to DROP, the appliance silently drops the request without responding to the client in any way. This action is appropriate for requests that appear to be part of a DDoS attack or other sustained attack on your servers.

Note: UNDEF events are triggered only for client requests. No UNDEF events are triggered for responses.

At the command prompt, type the following command to set the undefined action and verify the configuration:

- set responder param -undefAction (RESET | DROP | NOOP)
- show responder param

Example

```
>set responder param -undefAction RESET
 Done
> show responder param
      Action Name: RESET
 Done
>
```

1. Navigate to AppExpert > Responder, and then under Settings, click the Change Responder Settings link.
2. In the Set Responder Params dialog box, under Global Undefined-Result Action, select NOOP, RESET, or DROP.
3. Click OK. A message appears in the status bar, stating that the Responder Parameters have been configured.

Responder Action and Policy Examples

Nov 24, 2014

Responder actions and policies are powerful and complex, but you can get started with relatively simple applications. For typical examples, see "Example: Blocking Access from Specified IPs" and "Example: Redirecting a Client to a new URL."

The following procedures block access to your protected Web site(s) by clients originating from the CIDR 222.222.0.0/16. The responder sends an error message stating that the client is not authorized to access the URL requested.

To block access by using the command line interface

At the command prompt, type the following commands to block access:

- add responder action act_unauthorized respondwith '"HTTP/1.1 200 OK\r\n\r\n" + "Client: " + CLIENT.IP.SRC + " is not authorized to access URL:" + "HTTP.REQ.URL.HTTP_URL_SAFE"'
- add responder policy pol_un "CLIENT.IP.SRC.IN_SUBNET (222.222.0.0/16)" act_unauthorized
- bind responder global pol_un 10

To block access by using the configuration utility

1. In the navigation pane, expand Responder, and then click Actions.
2. In the details pane, click Add.
3. In the Create Responder Action dialog box, do the following:
   1. In the Name text box, type act_unauthorized.
   2. Under Type, select Respond with.
   3. In the Target text area, type the following string: "HTTP/1.1 200 OK\r\n\r\n" + "Client: " + CLIENT.IP.SRC + " is not authorized to access URL:" + HTTP.REQ.URL.HTTP_URL_SAFE
   4. Click Create, and then click Close.

   The responder action you configured, named act_unauthorized, now appears in the Responder Actions page.
4. In the navigation pane, click Policies.
5. In the details pane, click Add.
6. In the Create Responder Policy dialog box, do the following:
   1. In the Name text box, type pol_unauthorized.
   2. Under Action, select act_unauthorized.
   3. In the Expression window, type the following rule: CLIENT.IP.SRC.IN_SUBNET(222.222.0.0/16)
   4. Click Create, then click Close.

   The responder policy you configured, named pol_unauthorized, now appears in the Responder Policies page.
7. Globally bind your new policy, pol_unauthorized, as described in "Binding a Responder Policy."

The following procedures redirect clients who access your protected Web site(s) from within the CIDR 222.222.0.0/16 to a specified URL.

To redirect clients by using the command line interface

At the command prompt, type the following commands to redirect clients and verify the configuration:

- add responder action act_redirect redirect '"http://www.example.com/404.html"'
- show responder action act_redirect

- add responder policy pol_redirect "CLIENT.IP.SRC.IN_SUBNET(222.222.0.0/16)" act_redirect
- show responder policy pol_redirect
- bind responder global pol_redirect 10

Example

```
> add responder action act_redirect redirect '" http ://www.example.com/404.html "'
> add responder policy pol_redirect "CLIENT.IP.SRC.IN_SUBNET(222.222.0.0/16)" act_redirect
```

To redirect clients by using the configuration utility

1. Navigate to AppExpert > Responder > Actions.
2. In the details pane, click Add.
3. In the Create Responder Action dialog box, do the following:
   1. In the Name text box, type act_redirect.
   2. Under Type, select Redirect.
   3. In the Target text area, type the following string: "http://www.example.com/404.html"
   4. Click Create, then click Close.

   The responder action you configured, named act_redirect, now appears in the Responder Actions page.
4. In the navigation pane, click Policies.
5. In the details pane, click Add.
6. In the Create Responder Policy dialog box, do the following:
   1. In the Name text box, type pol_redirect.
   2. Under Action, select act_redirect.
   3. In the Expression window, type the following rule: CLIENT.IP.SRC.IN_SUBNET(222.222.0.0/16)
   4. Click Create, then click Close.

   The responder policy you configured, named pol_redirect, now appears in the Responder Policies page.
7. Globally bind your new policy, pol_redirect, as described in "Binding a Responder Policy."

Diameter Support for Responder

Apr 09, 2014

The Responder feature now supports the Diameter protocol. You can configure Responder to respond to Diameter requests as it does HTTP and TCP requests. For example, you could configure Responder to respond to requests from a specific Diameter origin with a redirect to a web page enhanced for mobile devices. A number of NetScaler expressions have been added that support examination of the Diameter header and the attribute-value pairs (AVPs). These expressions support lookup of specific AVPs by index, ID or name, examine the information in each AVP, and send an appropriate response.

To configure the Responder feature to send a response to a diameter request, at the command prompt, type the following commands:

- add responder action <actname> RESPONDWITH "DIAMETER.NEW_REDIRECT(\"aaa://host.example.com\")"
  For <actname>, substitute a name for your new action. The name can consist of from one to 127 characters in length, and can contain letters, numbers, and the hyphen (-) and underscore (_) symbols. For aaa://host.example.com, substitute the URL of the diameter host to which you want to redirect connections.
- add responder policy <polname> "diameter.req.avp(264).value.eq(\"host1.example.net\")" <actname>
  For <polname>, substitute a name for your new policy. As with <actname>, the name can consist of from one to 127 characters in length, and can contain letters, numbers, and the hyphen (-) and underscore (_) symbols. For host1.example.net, substitute the name of the originating host of the requests that you want to redirect. For <actname>, substitute the name of the action that you just created.
- bind lb vserver <vservername> -policyName <polname> -priority <priority> -type REQUEST
  For <vservername>, substitute the name of the load balancing virtual server to which you want to bind the policy. For <polname>, substitute the name of the policy you just created. For <priority>, substitute a priority for the policy.

Example

To create a Responder action and policy to respond to Diameter requests that originate from "host1.example.net" with a redirect to "host.example.com", you could add the following action and policy, and bind the policy as shown.

```
> add responder action act_resp-dm-redirect RESPONDWITH "DIAMETER.NEW_REDIRECT(\"aaa://host.example.com\")"
> add responder pol_resp-dm-redirect "diameter.req.avp(264).value.eq(\"host1.example.net\")" act_resp-dm-redirect
> bind lb vserver vs1 -policyName pol_resp-dm-redirect -priority 10 -type REQUEST
```

Done

Troubleshooting

Jul 22, 2013

If the responder feature does not work as expected after you have configured it, you can use some common tools to access NetScaler resources and diagnose the problem.

Resources for Troubleshooting

Updated: 2013-07-22

For best results, use the following resources to troubleshoot an integrated cache issue on a NetScaler appliance:

- The ns.conf file
- The relevant trace files from the client and the NetScaler appliance

In addition to the above resources, the following tools expedite troubleshooting:

- The iehttpheaders or a similar utility
- The Wireshark application customized for the NetScaler trace files

Troubleshooting Responder Issues

Updated: 2013-07-29

- **Issue**
  The Responder feature is configured, but the responder action is not working.

  Resolution
  - Verify that the feature is enabled.
  - Check the hit counters of any of the policies to see if the counters are getting incremented.
  - Verify that the policies and actions are configured correctly.
  - Verify that the actions and policies are bound appropriately.
  - Record the packet traces on the client and the NetScaler appliance, and analyze the them to get some pointer to the issue.
  - Record the iehttpHeaters packet traces on the client and verify the HTTP requests and responses to get some pointer to the issue.
- **Issue**
  You need to create a maintenance page.

  Resolution
  1. Configure the services and virtual Server.
  2. Configure a backup virtual server with a service bound to it. This ensures that the status of the Web site is always displayed as UP.
  3. Configure the primary virtual server to use the backup virtual server as a backup.
  4. Create a responder action with an appropriate target. Following is an example for your reference:
     `add responder action sorry_page respondwith q{"HTTP/1.0 200 OK" +"\r\n\r\n" + "<html> <body>Sorry, this page is not available</body></html>" + "\r\n"} .`
  5. Create a responder policy and bind the action to it.
  6. Bind the responder policy to the backup virtual Server.

Rewrite

Aug 30, 2013

Rewrite refers to the rewriting of some information in the requests or responses handled by the NetScaler appliance. Rewriting can help in providing access to the requested content without exposing unnecessary details about the Web site's actual configuration. A few situations in which the rewrite feature is useful are described below:

- To improve security, the NetScaler can rewrite all the http:// links to https:// in the response body.
- In the SSL offload deployment, the insecure links in the response have to be converted into secure links. Using the rewrite option, you can rewrite all the http:// links to https:// for making sure that the outgoing responses from NetScaler to the client have the secured links.
- If a Web site has to show an error page, you can show a custom error page instead of the default 404 Error page. For example, if you show the home page or site map of the Web site instead of an error page, the visitor remains on the site instead of moving away from the Web site.
- If you want to launch a new Web site, but use the old URL, you can use the Rewrite option.
- When a topic in a site has a complicated URL, you can rewrite it with a simple, easy-to-remember URL (also referred to as 'cool URL').
- You can append the default page name to the URL of a Web site. For example, if the default page of a company's Web site is 'http://www.abc.com/index.php', when the user types 'abc.com' in the address bar of the browser, you can rewrite the URL to 'abc.com/index.php'.

When you enable the rewrite feature, NetScaler can modify the headers and body of HTTP requests and responses.

To rewrite HTTP requests and responses, you can use protocol-aware NetScaler policy expressions in the rewrite policies you configure. The virtual servers that manage the HTTP requests and responses must be of type HTTP or SSL. In HTTP traffic, you can take the following actions:

- Modify the URL of a request
- Add, modify or delete headers
- Add, replace, or delete any specific string within the body or headers.

To rewrite TCP payloads, consider the payload as a raw stream of bytes. Each of the virtual servers that managing the TCP connections must be of type TCP or SSL_TCP. The term TCP rewrite is used to refer to the rewrite of TCP payloads that are not HTTP data. In TCP traffic, you can add, modify, or delete any part of the TCP payload.

For examples to use the rewrite feature, see "Rewrite Action and Policy Examples."

Comparison between Rewrite and Responder options

The main difference between the rewrite feature and the responder feature is as follows:

Responder cannot be used for response or server-based expressions. Responder can be used only for the following scenarios depending on client parameters:

- Redirecting a http request to new Web sites or Web pages
- Responding with some custom response
- Dropping or resetting a connection at request level

In case of a responder policy, the NetScaler examines the request from the client, takes action according to the applicable policies, sends the response to the client, and closes the connection with the client.

In case of a rewrite policy, the NetScaler examines the request from the client or response from the server, takes action according to the applicable policies, and forwards the traffic to the client or the server.

In general, it is recommended to use responder if you want the NetScaler to reset or drop a connection based on a client or request-based parameter. Use responder to redirect traffic, or respond with custom messages. Use rewrite for manipulating data on HTTP requests and responses.

How Rewrite Works

Aug 30, 2013

A rewrite policy consists of a rule and action. The rule determines the traffic on which rewrite is applied and the action determines the action to be taken by the NetScaler. You can define multiple rewrite policies. For each policy, specify the bind point and priority.

A bind point refers to a point in the traffic flow at which the NetScaler examines the traffic to verify whether any rewrite policy can be applied to it. You can bind a policy to a specific load balancing or content switching virtual server, or make the policy global if you want the policy to be applied to the entire traffic handled by the NetScaler. These policies are referred to as global policies.

In addition to the user-defined policies, the NetScaler has some default policies. You cannot modify or delete a default policy.

For evaluating the policies, NetScaler follows the order mentioned below:

- Global policies
- Policies bound to specific virtual servers
- Default policies

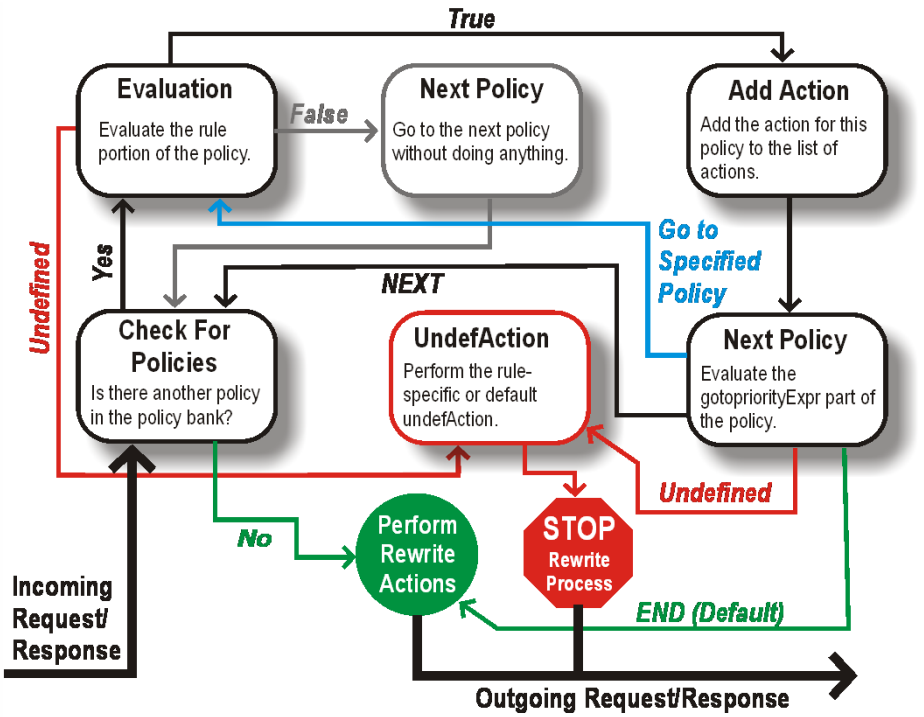
Note: NetScaler can apply a rewrite policy only when it is bound to a point.
NetScaler implements the rewrite feature in the following steps:

- The NetScaler appliance checks for global policies and then checks for policies at individual bind points.
- If multiple policies are bound to a bind point, the NetScaler evaluates the policies in the order of their priority. The policy with the highest priority is evaluated first. After evaluating each policy, if the policy is evaluated to TRUE (the traffic matches the rule), it adds the action associated with the policy to a list of actions to be performed. A match occurs when the characteristics specified in the policy rule match the characteristics of the request or response being evaluated.
- For any policy, in addition to the action, you can specify the policy that should be evaluated after the current policy is evaluated. This policy is referred to as the 'Go to Expression'. For any policy, if a Go to Expression (gotoPriorityExpr) is specified, the NetScaler evaluates the Go to Expression policy; it ignores policy with the next highest priority.
  You can specify the priority of the policy to indicate the Go to Expression policy; you cannot use the name of the policy.
  If you want the NetScaler to stop evaluating other policies after evaluating a particular policy, you can set the Go to Expression to 'END'.
- After all the policies are evaluated or when a policy has the Go to Expression set as END, the NetScaler starts performing the actions according to the list of actions.

For more information about configuring rewrite policies, see "Configuring a Rewrite Policy" and about binding rewrite policies, see "Binding a Rewrite Policy."

The following figure illustrates how NetScaler processes a request or response when the rewrite feature is used.

Figure 1. The Rewrite Process



Policy Evaluation

The policy with the highest priority is evaluated first. NetScaler does not stop the evaluation of rewrite policies when it finds a match; it evaluates all the rewrite policies configured on the NetScaler.

- If a policy evaluates to TRUE, the NetScaler follows the procedure below:
  - If the policy has the Go to Expression set to END, the NetScaler stops evaluating all the other policies and starts performing the rewrite.
  - The gotoPriorityExpression can be set to 'NEXT', 'END', some integer or 'INVOCATION_LIST'. The value determines the policy with the next priority. The following table shows the action taken by NetScaler for each value of the expression.

Value of the expression	Action
NEXT	Policy with the next priority gets evaluated.
END	Evaluation of policies stops.
<an integer>	Policy with specified priority gets evaluated.
INVOCATION_LIST	Goto NEXT or END is applied based on the result of the invocation list.

- If a policy evaluates to FALSE, the NetScaler continues the evaluation in the order of priority.
- If a policy evaluates to UNDEFINED (cannot be evaluated on the received traffic due to an error), the NetScaler performs the action assigned to the UNDEFINED condition (referred to as undefAction) and stops further evaluation of polices.

The NetScaler starts the actual rewriting only after the evaluation is complete. It refers to the list of actions identified by

policies that are evaluated to TRUE, and starts the rewriting. After implementing all the actions in the list, the NetScaler forwards the traffic as required.

Note: Ensure that the policies do not specify conflicting or overlapping actions on the same part of the HTTP header or body, or TCP payload. When such a conflict occurs, the NetScaler encounters an undefined situation and aborts the rewrite.

Rewrite Actions

On the NetScaler appliance, specify the actions to be taken such as adding, replacing, or deleting text within the body, or adding, modifying or deleting headers, or any changes in the TCP payload as rewrite actions. For more information about rewrite actions, see "Configuring a Rewrite Action."

The following table describes the steps the NetScaler can take when a policy evaluates to TRUE.

Action	Result
Insert	The rewrite action specified for the policy is carried out.
NOREWRITE	The request or response is not rewritten. NetScaler forwards the traffic without rewriting any part of the message.
RESET	The connection is aborted at the TCP level.
DROP	The message is dropped.

Note: For any policy, you can configure the undefaction (action to be taken when the policy evaluates to UNDEFINED) as NOREWRITE, RESET, or DROP.

To use the Rewrite feature, take the following steps:

- Enable the feature on the NetScaler.
- Define rewrite actions.
- Define rewrite policies.
- Bind the policies to a bind point to bring a policy into effect.

Enabling the Rewrite Feature

Aug 30, 2013

Enable the rewrite feature on the NetScaler appliance if you want to rewrite the HTTP or TCP requests or responses. If the feature is enabled, NetScaler takes rewrite action according to the specified policies. For more information, see "How Rewrite Works."

To enable the rewrite feature by using the command line interface

At the command prompt, type the following commands to enable the rewrite feature and verify the configuration:

- enable ns feature REWRITE
- show ns feature

Example

```
> enable ns feature REWRITE
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection         SP              ON
 .
 .
 .
 19)   Rewrite                  REWRITE         ON
 .
 .
 24)   NetScaler Push           push            OFF
 Done
```

To enable the rewrite feature by using the configuration utility

1. In the navigation pane, click System, and then click Settings.
2. In the details pane, under Modes and Features, click Configure basic features.
3. In the Configure Basic Features dialog box, select the Rewrite check box, and then click OK.
4. In the Enable/Disable Feature(s) dialog box, click Yes. A message appears in the status bar, stating that the selected feature was enabled.

Configuring a Rewrite Action

Nov 24, 2014

After enabling the rewrite feature, you need to configure one or more actions unless a built-in rewrite action is sufficient. All of the built-in actions have names beginning with the string ns_cvpn, followed by a string of letters and underscore characters. Built-in actions perform useful and complex tasks such as decoding parts of a clientless VPN request or response or modifying JavaScript or XML data. The built-in actions can be viewed, enabled, and disabled, but cannot be modified or deleted.

Target expressions in actions for TCP rewrite must begin with one of the following expression prefixes:

- **CLIENT.TCP.PAYLOAD.** For rewriting TCP payloads in client requests. For example, CLIENT.TCP.PAYLOAD(10000).AFTER_STR("string1").
- **SERVER.TCP.PAYLOAD.** For rewriting TCP payloads in server responses. For example, SERVER.TCP.PAYLOAD(1000).B64DECODE.BETWEEN("string1","string2").

You can use all types of existing string manipulation functions with these prefixes to identify the strings that you want to rewrite. To configure a rewrite action, you assign it a name, specify an action type, and add one or more arguments specifying additional data. The following table describes the action types and the arguments you use with them.
Note: Action types that can be used only for HTTP rewrite are identified in the **Rewrite Action Type** column.

Table 1. Rewrite Action Types and Their Arguments

Rewrite Action Type	Argument 1	Argument 2
INSERT_HTTP_HEADER: Inserts the HTTP header you specify into the HTTP request or response. This is the default choice. This action type can be used only with HTTP requests and responses.	The HTTP header you want to insert. For example, if you want to insert the client IP from which a request is sent, type Client-IP.	A string expression that describes the contents of the header you want to insert. For example, if you want to insert the Client IP from which a request is sent, type CLIENT.IP.SRC.
INSERT_BEFORE: Inserts a new string before the designated string.	A string expression that describes the string before which you want to insert a new string. For example, if you want to find the hostname www.example.com and insert a string before the example.com portion, type the following: HTTP.REQ.HOSTNAME.BEFORE_STR ("example.com")	A string expression that describes the new string you want to insert. For example, if you want to insert the new string en. before the string example in the hostname, type en followed by a period.
INSERT_AFTER: Inserts a new string after the designated string.	A string expression that describes the string after which you want to insert a	A string expression that describes the new string

Rewrite Action Type	Argument 1	Argument 2
	new string. For example, if you want to find the hostname www.example.com, and insert a string after the www. portion, type the following: HTTP.REQ.HOSTNAME.AFTER_STR ("www.")	you want to insert. For example, if you want to insert the new string en. after the string www. in the hostname, type en followed by a period.
REPLACE: Replaces the designated string with a different string.	A string expression that describes the string you want to replace with a new string. For example, if you want to replace the entire hostname in the Host header, type HTTP.REQ.HOSTNAME.SERVER.	A string expression that describes the new string you want to insert. For example, if you want to replace the current host header with the string web01.example.net, type web01.example.net.
DELETE: Deletes the designated string.	A string expression that describes the string you want to delete. For example, if you want to find and delete the string .en in the hostname of HTTP response headers, type the following: HTTP.RES.HEADER("Host").SUBSTR("en.")	
DELETE_HTTP_HEADER: Deletes the designated HTTP header, including all header contents. This action type can be used only with HTTP requests and responses.	The name of the HTTP header you want to delete. For example, if you want to delete the cache-control header from HTTP responses, type HTTP.RES.HEADER ("Cache-Control").	
CORRUPT_HTTP_HEADER: Replaces the name of the given HTTP header with a corrupted name so that it will not be recognized by the receiver. This action type can be used only with HTTP requests and responses.	The name of the HTTP header that you want to corrupt. If the specified header occurs more than once in a request, all the occurrences are corrupted. For example, if you want to corrupt the Host header in an HTTP request, you can use the following rewrite action command: add rewrite action corrupt_header_act	

Rewrite Action Type	Argument 1	Argument 2
	CORRUPT_HTTP_HEADER Host.	
REPLACE_HTTP_RES: Replace the http response with the value specified in the target field. This action type can be used only with HTTP requests and responses.	A string expression that describes the string you want to replace the HTTP response with. For example, type HTTP 200 OK You are not authorized to view this page to replace the entire HTTP response with this warning.	
REPLACE_ALL: Will replace all occurrences of a pattern in the target text reference with the value specified in the string builder expression.	The part of either the HTTP request or response where you want to carry out the replacement.	A string expression that describes the new string you want to insert.
DELETE_ALL: Delete every occurrence of the pattern specified in the target text reference.	The part of either the HTTP request or response where you want the deletion to occur.	A string pattern after which the deletion should occur.
INSERT_AFTER_ALL: Inserts the value specified by string builder expression after each occurrence of a specified pattern in the target text reference.	The part of either the HTTP request or response where you want the insertion to occur.	A string expression that describes the new string you want to insert.
INSERT_BEFORE_ALL: Inserts the value you specify before each occurrence of the pattern you specify.	The part of either the HTTP request or response that you want to delete.	A string expression that describes the new string you want to insert.
CLIENTLESS_VPN_ENCODE: Encodes the URL you specify in clientless VPN format.	The URL you want to encode.	
CLIENTLESS_VPN_ENCODE_ALL: Encodes all of the URLs you specify in clientless VPN format.	A pattern that matches the URLs you want to encode.	
CLIENTLESS_VPN_DECODE: Decodes the URL you specify from clientless VPN format and returns it as unencoded text.	The URL you want to decode.	
CLIENTLESS_VPN_DECODE_ALL: Decodes all of the URLs you specify from clientless VPN format and returns them as unencoded text.	A pattern that matches all of the URLs you want to decode.	

To create a new rewrite action by using the command line interface

At the command prompt, type the following commands to create a new rewrite action and verify the configuration:

- add rewrite action <name> <type> <target> [<stringBuilderExpr>] [(-pattern <expression> | -patset <string>)] [-bypassSafetyCheck (YES | NO)]
- show rewrite action <name>

Example 1: Inserting an HTTP Header With the Client IP

```
> add rewrite action insertact INSERT_HTTP_HEADER "client-IP" CLIENT.IP.SRC
Done

> show rewrite action insertact

      Name: insertact
      Operation: insert_http_header   Target:Client-IP
      Value:CLIENT.IP.SRC
      BypassSafetyCheck : NO
      Hits: 0
      Undef Hits: 0
      Action Reference Count: 0
 Done
```

Example 2: Replacing Strings in a TCP Payload (TCP Rewrite)

```
> add rewrite action client_tcp_payload_replace_all REPLACE_ALL
  'client.tcp.payload(1000)' '"new-string"' -search text("old-string")
 Done
> show rewrite action client_tcp_payload_replace_all

      Name: client_tcp_payload_replace_all
      Operation: replace_all
      Target:client.tcp.payload(1000)
      Value:"new-string"
      Search: text("old-string")
      BypassSafetyCheck : NO
      Hits: 0
      Undef Hits: 0
      Action Reference Count: 0
 Done
>
```

To modify an existing rewrite action by using the command line interface

At the command prompt, type the following commands to modify an existing rewrite action and verify the configuration:

- set rewrite action <name> [-target <string>] [-stringBuilderExpr <string>] [(-pattern <expression> | -patset <string>)] [-bypassSafetyCheck (YES | NO)]
- show rewrite action <name>

Example

```
> set rewrite action insertact -target "Client-IP"
 Done
```

```
> show rewrite action insertact

      Name: insertact
      Operation: insert_http_header   Target:Client-IP
      Value:CLIENT.IP.SRC
      BypassSafetyCheck : NO
      Hits: 0
      Undef Hits: 0
      Action Reference Count: 0
 Done
```

To remove a rewrite action by using the command line interface

At the command prompt, type the following commands to remove a rewrite action :
rm rewrite action <name>
Example

```
> rm rewrite action insertact
Done
```

To configure a rewrite action by using the configuration utility

1. Navigate to AppExpert > Rewrite > Actions.
2. In the details pane, do one of the following:
   - To create a new action, click Add.
   - To modify an existing action, select the action, and then click Open.
3. Click Create or OK. A message appears in the status bar, stating that the Action has been configured successfully.
4. Repeat steps 2 through 4 to create or modify as many rewrite actions as you wish.
5. Click Close.

To add an expression by using the Add Expression dialog box

1. In the Create Rewrite Action or Configure Rewrite Action dialog box, under the text area for the type argument you want to enter, click Add.
2. In the Add Expression dialog box, in the first list box choose the first term for your expression.
   HTTP
   The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.
   SYS
   The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   CLIENT
   The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.

   When you make your choice, the rightmost list box lists appropriate terms for the next part of your expression.
3. In the second list box, choose the second term for your expression. The choices depend upon which choice you made in the previous step, and are appropriate to the context. After you make your second choice, the Help window below the Construct Expression window (which was blank) displays help describing the purpose and use of the term you just chose.
4. Continue choosing terms from the list boxes that appear to the right of the previous list box, or typing strings or numbers in the text boxes that appear to prompt you to enter a value, until your expression is finished.

For more information about the PI expressions language and creating expressions for responder policies, see "Policies and Expressions."

If you want to test the effect of a rewrite action when used on sample HTTP data, you can use the Rewrite Expression Evaluator.
Note: The Rewrite Expression Evaluator is only available in the configuration utility. There is no NetScaler command line version.

To evaluate a rewrite action by using the Rewrite Action Evaluator dialog box

1. In the Rewrite Actions details pane, select the rewrite action that you want to evaluate, and then click Evaluate.
2. In the Rewrite Expression Evaluator dialog box, specify values for the following parameters. (An asterisk indicates a required parameter.)
   - Rewrite Action*—If the rewrite action you want to evaluate is not already selected, select it from the drop-down list. After you select a Rewrite action, the Details section displays the details of the selected Rewrite action.
   - New*—Select New to open the Create Rewrite Action dialog box and create a new rewrite action.
   - Modify*—Select Modify to open the Configure Rewrite Action dialog box and modify the selected rewrite action.
   - Flow Type*—Specifies whether to test the selected rewrite action with HTTP Request data or HTTP Response data. The default is Request. If you want to test with Response data, select Response.
   - HTTP Request/Response Data*—Provides a space for you to provide the HTTP data that the Rewrite Action Evaluator will use for testing. You can paste the data directly into the window, or click Sample to insert some sample HTTP headers.
   - Show end-of-line—Specifies whether to show UNIX-style end-of-line characters (\n) at the end of each line of sample HTTP data.
   - Sample—Inserts sample HTTP data into the HTTP Request/Response Data window. You can choose either GET or POST data.
   - Browse—Opens a local browse window so that you can choose a file containing sample HTTP data from a local or network location.
   - Clear—Clears the current sample HTTP data from the HTTP Request/Response Data window.
3. Click Evaluate. The Rewrite Action Evaluator evaluates the effect of the Rewrite action on the sample data that you chose, and displays the results as modified by the selected Rewrite action in the Results window. Additions and deletions are highlighted as indicated in the legend in the lower left-hand corner of the dialog box.
4. Continue evaluating Rewrite actions until you have determined that all of your actions have the effect that you wanted.
   - You can modify the selected rewrite action and test the modified version by clicking Modify to open the Configure Rewrite Action dialog box, making and saving your changes, and then clicking Evaluate again.
   - You can evaluate a different rewrite action using the same request or response data by selecting it from the Rewrite Action drop-down list, and then clicking Evaluate again.
5. Click Close to close the Rewrite Expression Evaluator and return to the Rewrite Actions pane.

To delete a rewrite action, select the rewrite action you want to delete, then click Remove and, when prompted, confirm your choice by clicking OK.

Configuring a Rewrite Policy

Aug 30, 2013

After you create any needed rewrite action(s), you must create at least one rewrite policy to select the requests that you want the NetScaler appliance to rewrite.

A rewrite policy consists of a rule, which itself consists of one or more expressions, and an associated action that is performed if a request or response matches the rule. Policy rules for evaluating HTTP requests and responses can be based on almost any part of a request or response.

Even though you cannot use TCP rewrite actions to rewrite data other than the TCP payload, you can base the policy rules for TCP rewrite policies on the information in the transport layer and the layers below the transport layer.

If a configured rule matches a request or response, the corresponding policy is triggered and the action associated with it is carried out.

Note: You can use either the command line interface or the configuration utility to create and configure rewrite policies. Users who are not thoroughly familiar with the command line interface and the NetScaler Policy expression language will usually find using the configuration utility much easier.

To add a new rewrite policy by using the command line interface

At the command prompt, type the following commands to add a new rewrite policy and verify the configuration:

- add rewrite policy <name> <expression> <action> [<undefaction>]
- show rewrite policy <name>

Example 1: Rewriting HTTP Content

```
> add rewrite policy policyNew "HTTP.RES.IS_VALID" insertact NOREWRITE
 Done
> show rewrite policy policyNew
      Name: policyNew
      Rule: HTTP.RES.IS_VALID
      RewriteAction: insertact
      UndefAction: NOREWRITE
      Hits: 0
      Undef Hits: 0

 Done
```

Example 2: Rewriting a TCP Payload (TCP Rewrite)

```
> add rewrite policy client_tcp_payload_policy CLIENT.IP.SRC.EQ(172.168.12.232) client_tcp_payload_replace_all
 Done
> show rewrite policy client_tcp_payload_policy
      Name: client_tcp_payload_policy
      Rule: CLIENT.IP.SRC.EQ(172.168.12.232)
      RewriteAction: client_tcp_payload_replace_all
      UndefAction: Use Global
      LogAction: Use Global
      Hits: 0
      Undef Hits: 0

 Done
>
```

To modify an existing rewrite policy by using the command line interface

At the command prompt, type the following commands to modify an existing rewrite policy and verify the configuration:

- set rewrite policy <name> -rule <expression> -action <action> [<undefaction>]
- show rewrite policy <name>

Example

```
> set rewrite policy policyNew -rule "HTTP.RES.IS_VALID" -action insertaction
 Done

> show rewrite policy policyNew
      Name: policyNew
      Rule: HTTP.RES.IS_VALID
      RewriteAction: insertaction
      UndefAction: NOREWRITE
      Hits: 0
      Undef Hits: 0

 Done
```

To remove a rewrite policy by using the command line interface

At the command prompt, type the following command to remove a rewrite policy:
rm rewrite policy <name>

Example

```
> rm rewrite policy policyNew
Done
```

To configure a rewrite policy by using the configuration utility

1. Navigate to AppExpert > Rewrite > Policies.
2. In the details pane, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the policy, and then click Open.
3. Click Create or OK. A message appears in the status bar, stating that the Policy has been configured successfully.
4. Repeat steps 2 through 4 to create or modify as many rewrite actions as you wish.
5. Click Close. To delete a rewrite policy, select the rewrite policy you want to delete, then click Remove and, when prompted, confirm your choice by clicking OK.

Binding a Rewrite Policy

Oct 29, 2013

After creating a rewrite policy, you must bind it to put it into effect. You can bind your policy to Global if you want to apply it to all traffic that passes through your NetScaler, or you can bind your policy to a specific virtual server or bind point to direct only that virtual server or bind point's incoming traffic to that policy. If an incoming request matches a rewrite policy, the action associated with that policy is carried out.

Rewrite policies for evaluating HTTP requests and responses can be bound to virtual servers of type HTTP or SSL, or they can be bound to the REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, and RES_DEFAULT bind points. Rewrite policies for TCP rewrite can be bound only to virtual servers of type TCP or SSL_TCP, or to the OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, OTHERTCP_RES_OVERRIDE, and OTHERTCP_RES_DEFAULT bind points.
Note: The term OTHERTCP is used in the context of the NetScaler appliance to refer to all TCP or SSL_TCP requests and responses that you want to treat as a raw stream of bytes regardless of the protocols that the TCP packets encapsulate.
When you bind a policy, you assign it a priority. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer.

In the NetScaler operating system, policy priorities work in reverse order - the higher the number, the lower the priority. For example, if you have three policies with priorities of 10, 100, and 1000, the policy assigned a priority of 10 is applied first, then the policy assigned a priority of 100, and finally the policy assigned an order of 1000.

Unlike most other features in the NetScaler operating system, the rewrite feature continues to evaluate and implement policies after a request matches a policy. However, the effect of a particular action policy on a request or response will often be different depending on whether it is performed before or after another action. Priority is important to get the results you intended.

You can leave yourself plenty of room to add other policies in any order, and still set them to evaluate in the order you want, by setting priorities with intervals of 50 or 100 between each policy when you bind it. If you do this, you can add additional policies at any time without having to reassign the priority of an existing policy.

When binding a rewrite policy, you also have the option of assigning a goto expression (gotoPriorityExpression) to the policy. A goto expression can be any positive integer that matches the priority assigned to a different policy that has a higher priority than the policy that contains the goto expression. If you assign a goto expression to a policy, and a request or response matches the policy, the NetScaler will immediately go to the policy whose priority matches the goto expression. It will skip over any policies with priority numbers that are lower than that of the current policy, but higher than the priority number of the goto expression, and not evaluate those policies.

For more information about binding policies on the NetScaler, see "Binding a Rewrite Policy."

To globally bind a rewrite policy by using the command line interface

At the command prompt, type the following commands to globally bind a rewrite policy and verify the configuration:

- bind rewrite global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]
- show rewrite global

Example

```
>bind rewrite global policyNew 10
 Done

> show rewrite global
1)     Global bindpoint: RES_DEFAULT
       Number of bound policies: 1

2)     Global bindpoint: REQ_OVERRIDE
       Number of bound policies: 1

 Done
```

To bind rewrite policy to a specific virtual server by using the command line interface

At the command prompt, type the following commands to bind rewrite policy to a specific virtual server and verify the configuration:

- bind lb vserver <name>@ (<serviceName>@ [-weight <positive_integer>]) | <serviceGroupName>@ | (-policyName <string>@ [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type (REQUEST | RESPONSE)] [-invoke (<labelType> <labelName>)])
- show lb vserver <name>

Example

```
> bind lb vserver lbvip -policyName ns_cmp_msapp -priority 50
 Done
>
> show lb vserver lbvip
       lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
       State: DOWN
       Last state change was at Wed Jul 15 05:54:24 2009 (+226 ms)
       Time since last state change: 28 days, 01:57:26.350
       Effective State: DOWN
       Client Idle Timeout: 180 sec
       Down state flush: ENABLED
       Disable Primary Vserver On Down : DISABLED
       Port Rewrite : DISABLED
       No. of Bound Services :  0 (Total)      0 (Active)
       Configured Method: LEASTCONNECTION
       Mode: IP
       Persistence: NONE
       Vserver IP and Port insertion: OFF
       Push: DISABLED  Push VServer:
       Push Multi Clients: NO
       Push Label Rule: none

1)     Policy : ns_cmp_msapp Priority:50
2)     Policy : cf-pol Priority:1      Inherited
 Done
```

To bind a rewrite policy to a bind point by using the configuration utility

1. Navigate to AppExpert > Rewrite > Policies.
2. In the details pane, select the rewrite policy you want to globally bind, and then click Policy Manager.
3. In the Rewrite Policy Manager dialog box, in the Bind Points menu, do one of the following:
    1. If you want to configure bindings for HTTP rewrite policies, click HTTP, and then click either Request or Response, depending on whether you want to configure request-based rewrite policies or response-based rewrite policies.
    2. If you want to configure bindings for TCP rewrite policies, click TCP, and then click either Client or Server, depending on whether you want to configure client-side TCP rewrite policies or server-side TCP rewrite policies.
4. Click the bind point to which you want to bind the rewrite policy. The Rewrite Policy Manager dialog box displays all the rewrite policies that are bound to the selected bind point.
5. Click Insert Policy to insert a new row and display a drop-down list with all available, unbound rewrite policies.
6. Click the policy you want to bind to the bind point. The policy is inserted into the list of rewrite policies bound to the bind point.
7. In the Priority column, you can change the priority to any positive integer. For more information about this parameter, see priority in "Parameters for binding a rewrite policy."
8. If you want to skip over policies and go directly to a specific policy in the event that the current policy is matched, change the value in the Goto Expression column to equal the priority of the next policy to be applied.. For more information about this parameter, see gotoPriorityExpression in "Parameters for binding a rewrite policy."
9. To modify a policy, click the policy, and then click Modify Policy.
10. To unbind a policy, click the policy, and then click Unbind Policy.
11. To modify an action, in the Action column, click the action you want to modify, and then click Modify Action.
12. To modify an invoke label, in the Invoke column, click the invoke label you want to modify, and then click Modify Invoke Label.
13. To regenerate the priorities of all the policies that are bound to the bind point you are currently configuring, click Regenerate Priorities. The policies retain their existing priorities relative to the other policies, but the priorities are renumbered in multiples of ten.
14. Click Apply Changes.
15. Click Close. A message appears in the status bar, stating that the Policy has been configured successfully.

To bind a rewrite policy to a specific virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane list of virtual servers, select the virtual server to which you want to bind the rewrite policy, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, select the Policies tab. All policies configured on your NetScaler appear on the list.
4. Select the check box next to the name of the policy you want to bind to this virtual server.
5. Click OK. A message appears in the status bar, stating that the Policy has been configured successfully.

Configuring Rewrite Policy Labels

Aug 30, 2013

If you want to build a more complex policy structure than is supported by single policies, you can create policy labels and then bind them as you would policies. A policy label is a user-defined point to which policies are bound. When a policy label is invoked, all the policies bound to it are evaluated in the order of the priority you configured. A policy label can include one or multiple policies, each of which can be assigned its own result. A match on one policy in the policy label can result in proceeding to the next policy, invoking a different policy label or appropriate resource, or an immediate end to policy evaluation and return of control to the policy that invoked the policy label.

A rewrite policy label consists of a name, a transform name that describes the type of policy included in the policy label, and a list of policies bound to the policy label. Each policy that is bound to the policy label contains all of the elements described in "Configuring a Rewrite Policy."

Note: You can use either the command line interface or the configuration utility to create and configure rewrite policy labels. Users who are not thoroughly familiar with the command line interface and the NetScaler Policy Infrastructure (PI) language will usually find using the configuration utility much easier.

To configure a rewrite policy label by using the command line interface

To add a new rewrite policy label, at the command prompt, type the following command:

```
add rewrite policylabel <labelName> <transform>
```

For example, to add a rewrite policy label named **polLabelHTTPResponses** to group all policies that work on HTTP responses, you would type the following:

```
add rewrite policylabel polLabelHTTPResponses http_res
```

To modify an existing rewrite policy label, at the NetScaler command prompt, type the following command:

```
set rewrite policy <name> <transform>
```

Note: The set rewrite policy command takes the same options as the add rewrite policy command.
To remove a rewrite policy label, at the NetScaler command prompt, type the following command:

```
rm rewrite policy<name>
```

For example, to remove a rewrite policy label named **polLabelHTTPResponses**, you would type the following:

```
rm rewrite policy polLabelHTTPResponses
```

To configure a rewrite policy label by using the configuration utility

1. Navigate to AppExpert > Rewrite > Policy Labels.
2. In the details pane, do one of the following:
   - To create a new policy label, click Add.
   - To modify an existing policy label, select the policy, and then click Open.
3. Add or remove policies from the list that is bound to the policy label.
   - To add a policy to the list, click Insert Policy, and choose a policy from the drop-down list. You can create a new policy and add it to the list by choosing New Policy in the list, and following the instructions in "Configuring a Rewrite Policy."

- To remove a policy from the list, select that policy, and then click Unbind Policy.

4. Modify the priority of each policy by editing the number in the Priority column.

   You can also automatically renumber policies by clicking Regenerate Priorities.

5. Click Create or OK, and then click Close.

   To remove a policy label, select it, and then click Remove. To rename a policy label, select it and then click Rename. Edit the name of the policy, and then click OK to save your changes.

Configuring the Default Rewrite Action

Aug 30, 2013

An undefined event is triggered when the NetScaler cannot evaluate a policy, usually because it detects a logical or other error in the policy or an error condition on the NetScaler. When the rewrite policy evaluation results in an error, the specified undefined action is carried out. Undefined actions configured at the rewrite policy level are carried out before a globally configured undefined action.

The NetScaler supports following three types of undefined actions:

undefAction NOREWRITE

Aborts rewrite processing, but does not alter the packet flow. This means that the NetScaler continues to process requests and responses that do not match any rewrite policy, and eventually forwards them to the requested URL unless another feature intervenes and blocks or redirects the request. This action is appropriate for normal requests to your Web servers, and is the default setting.

undefAction RESET

Resets the client connection. This means that the NetScaler tells the client that it must re-establish its session with the Web server. This action is appropriate for repeat requests for Web pages that do not exist, or for connections that might be attempts to hack or probe your protected Web site(s).

undefAction DROP

Silently drops the request without responding to the client in any way. This means that the NetScaler simply discards the connection without responding to the client. This action is appropriate for requests that appear to be part of a DDoS attack or another sustained attack on your servers.

Note: Undefined events can be triggered for both request and response flow specific policies.

To configure the default action by using the command line interface

At the command prompt, type the following commands to configure the default action and verify the configuration:

- set rewrite param -undefAction (NOREWRITE | RESET | DROP)
- show rewrite param

Example

```
> set rewrite param -undefAction NOREWRITE
 Done
> show rewrite param
      Action Name: NOREWRITE
 Done
```

To configure the default action by using the configuration utility

1. Navigate to AppExpert > Rewrite.
2. In the details pane, under Rewrite Overview, click the Change Rewrite Settings link. The Set Rewrite Params dialog box appears.
3. Under Global Undefined-Result Action, select an option as follows:
   - NoRewrite—NOREWRITE
   - Reset—RESET
   - Drop—DROP

4. Click OK. The global undefined action is set to the value you chose.

Bypassing the Safety Check

Oct 29, 2013

When you create a rewrite action, the NetScaler verifies that the expression you used to create the action is safe. Expressions created by the NetScaler from run-time data, such as URLs contained in HTTP requests, can cause unexpected errors. The NetScaler reports expressions that cause such errors as unsafe expressions.

In some cases, the expressions may be safe. For example, the NetScaler cannot validate an expression that contains a URL that does not resolve, even if the URL does not resolve because the Web server is temporarily unavailable. You can manually bypass the Safety Check to allow these expressions.

To bypass the safety check by using the command line interface

At the command prompt, type the following commands to bypass the safety check and verify the configuration:

- set rewrite action <name> -bypassSafetyCheck YES
- show rewrite action <name>

Example

```
> set rewrite action insertact -bypassSafetyCheck YES
 Done
> show rewrite action insertact

      Name: insertact
      Operation: insert_http_header   Target:Client-IP
      Value:CLIENT.IP.SRC
      BypassSafetyCheck : YES
      Hits: 0
      Undef Hits: 0
      Action Reference Count: 2
 Done
```

To bypass safety check by using the configuration utility

1. Navigate to AppExpert > Rewrite > Actions.
2. In the details pane, select the rewrite action to be exempted from the safety check, and then click Open.
3. In the Configure Rewrite Action dialog box, select the Bypass Safety Check check box.
4. Click OK.

Rewrite Action and Policy Examples

Mar 20, 2012

The examples in this section demonstrate how to configure rewrite to perform various useful tasks. The examples occur in the server room of Example Manufacturing Inc., a mid-sized manufacturing company that uses its Web site to manage a considerable portion of its sales, deliveries, and customer support.

Example Manufacturing has two domains: example.com for its Web site and email to customers, and example.net for its intranet. Customers use the Example Web site to place orders, request quotes, research products, and contact customer service and technical support.

As an important part of Example's revenue stream, the Web site must respond quickly and keep customer data confidential. Example therefore has several Web servers and uses Citrix NetScaler appliances to balance the Web site load and manage traffic to and from its Web servers.

The Example system administrators use the rewrite features to perform the following tasks:

Example 1: Delete old X-Forwarded-For and Client-IP Headers.

Example Inc. removes old X-Forwarded-For and Client-IP HTTP headers from incoming requests.

Example 2: Adding a Local Client-IP Header.

Example Inc. adds a new, local Client-IP header to incoming requests.

Example 3: Tagging Secure and Insecure Connections.

Example Inc. tags incoming requests with a header that indicates whether the connection is a secure connection.

Example 4: Mask the HTTP Server Type.

Example Inc. modifies the HTTP Server: header so that unauthorized users and malicious code cannot use that header to determine the HTTP server software it uses.

Example 5: Redirect an External URL to an Internal URL.

Example Inc. hides information about the actual names of its Web servers and the configuration of its server room from users, to make URLs on its Web site shorter and easier to remember, and to improve security on its site.

Example 6: Migrating Apache Rewrite Module Rules.

Example Inc. moved its Apache rewrite rules to a NetScaler appliance, translating the Apache PERL-based script syntax to the NetScaler rewrite rule syntax.

Example 7: Marketing Keyword Redirection.

The marketing department at Example Inc. sets up simplified URLs for certain predefined keyword searches on the company's Web site.

Example 8: Redirect Queries to the Queried Server.

Example Inc. redirects certain query requests to the appropriate server.

Example 9: Home Page Redirection.

Example Inc. recently acquired a smaller competitor, and it now redirects requests for the acquired company's home page to a page on its own Web site.

Each of these tasks requires that the system administrators create rewrite actions and policies and bind them to a valid bind point on the NetScaler.

Example 1: Delete Old X-Forwarded-For and Client-IP Headers

Aug 30, 2013

Example Inc. wants to remove old X-Forwarded-For and Client-IP HTTP headers from incoming requests, so that the only X-Forwarded-For headers that appear are the ones added by the local server. This configuration can be done through the NetScaler command line or the configuration utility. The Example Inc. system administrator is an old-school networking engineer and prefers to use a CLI where possible, but wants to be sure he understands the configuration utility interface so that he can show new system administrators on the team how to use it.

The examples below demonstrate how to perform each configuration with both the CLI and the configuration utility. The procedures are abbreviated on the assumption that users will already know the basics of creating rewrite actions, creating rewrite policies, and binding policies.

- For more detailed information about creating rewrite actions, see "Configuring a Rewrite Action."
- For more detailed information about creating rewrite policies, see "Configuring a Rewrite Policy."
- For more detailed information about binding rewrite policies, see "Binding a Rewrite Policy."

To delete old X-Forwarded and Client-IP headers from a request by using the command line interface

At the command prompt, type the following commands in the order shown:

```
add rewrite action act_del_xfor delete_http_header x-forwarded-for
add rewrite action act_del_cip delete_http_header client-ip
add rewrite policy pol_check_xfor 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS' act_del_xfor
add rewrite policy pol_check_cip 'HTTP.REQ.HEADER("client-ip").EXISTS' act_del_cip
bind rewrite global pol_check_xfor 100 200
bind rewrite global pol_check_cip 200 300
```

To delete old X-Forwarded and Client-IP headers from a request by using the configuration utility

In the Create Rewrite Action dialog box, create two rewrite actions with the following descriptions.

Name	Type	Argument(s)
act_del_xfor	delete_http_header	x-forwarded-for
act_del_cip	delete_http_header	client-ip

In the Create Rewrite Policy dialog box, create two rewrite policies with the following descriptions.

Name	Expression	Action
pol_check_xfor	'HTTP.REQ.HEADER("x-forwarded-for").EXISTS'	act_del_xfor
pol_check_cip	'HTTP.REQ.HEADER("client-ip").EXISTS'	act_del_cip

Bind both policies to global, assigning the priorities and goto expression values shown below.

Name	Priority	Goto Expression
pol_check_xfor	100	200
pol_check_xfor	200	300

All old X-Forwarded-For and Client-IP HTTP headers are now deleted from incoming requests.

Example 2: Adding a Local Client-IP Header

Aug 30, 2013

Example Inc. wants to add a local Client-IP HTTP header to incoming requests. This example contains two slightly different versions of the same basic task.

To add a local Client-IP header by using the command line interface

At the command prompt, type the following commands in the order shown:

```
add rewrite action act_ins_client insert_http_header NS-Client 'CLIENT.IP.SRC'
add rewrite policy pol_ins_client 'HTTP.REQ.HEADER("x-forwarded-for").EXISTS || HTTP.REQ.HEADER("client-ip").EXISTS' act_ins_client
bind rewrite global pol_ins_client 300 END
```

To add a local Client-IP header by using the configuration utility

In the Create Rewrite Action dialog box, create a rewrite action with the following description.

Name	Type	Argument(s)
act_ins_client	insert_http_header	NS-Client 'CLIENT.IP.SRC'

In the Create Rewrite Policy dialog box, create a rewrite policy with the following description.

Name	Expression	Action
pol_ins_client	'HTTP.REQ.HEADER("x-forwarded-for").EXISTS || HTTP.REQ.HEADER("client-ip").EXISTS'	act_ins_client

Bind both policies to global, assigning the priorities and goto expression values shown below.

Name	Priority	Goto Expression
pol_check_xfor	100	200
pol_check_xfor	200	300

A local Client-IP HTTP header is now added to incoming requests. You can also modify the configuration above to append all IPs from X-Forwarded-For headers to the new Client-IP header, as shown below.

Example 3: Tagging Secure and Insecure Connections

Mar 20, 2012

Example Inc. wants to tag incoming requests with a header that indicates whether or not the connection is a secure connection. This helps the server keep track of secure connections after the NetScaler has decrypted the connections.

To implement this configuration, you would begin by creating rewrite actions with the values shown in the following tables. These actions label connections to port 80 as insecure connections, and connections to port 443 as secure connections.

Action Name	Type of Rewrite Action	Header Name	Value
Action-Rewrite-SSL_YES	INSERT_HTTP_HEADER	SSL	YES

Action Name	Type of Rewrite Action	Header Name	Value
Action-Rewrite-SSL_NO	INSERT_HTTP_HEADER	SSL	NO

You would then create a rewrite policy with the values shown in the following tables. These policies check incoming requests to determine which requests are directed to port 80 and which are directed to port 443. The policies then add the correct SSL header.

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-SSL_YES	Action-Rewrite-SSL_YES	NOREWRITE	CLIENT.TCP.DSTPORT.EQ(443)
Policy-Rewrite-SSL_NO	Action-Rewrite-SSL_NO	NOREWRITE	CLIENT.TCP.DSTPORT.EQ(80)

Finally, you would bind the rewrite policies to NetScaler, assigning the first policy a priority of 200, and the second a priority of 300, and setting the goto expression of both policies to END.

Each incoming connection to port 80 now has an SSL:NO HTTP header added to it and each incoming connection to port 443 has an SSL:YES HTTP header added to it.

Example 4: Mask the HTTP Server Type

Mar 20, 2012

Example Inc. wants to modify the HTTP Server: header so that unauthorized users and malicious code cannot use the header to identify the software that the HTTP server uses.

To modify the HTTP Server: header, you would create a rewrite action and a rewrite policy with the values in the following tables.

Action Name	Type of Rewrite Action	Expression to choose target reference	String expression for replacement text
Action-Rewrite-Server_Mask	REPLACE	HTTP.RES.HEADER("Server")	"Web Server 1.0"

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-Server_Mask	Action-Rewrite-Server_Mask	NOREWRITE	HTTP.RES.IS_VALID

You would then globally bind the rewrite policy, assigning a priority of 100 and setting the Goto Priority Expression of the policy to END.

The HTTP Server: header is now modified to read “Web Server 1.0,” masking the actual HTTP server software used by the Example Inc. Web site.

Example 5: Redirect an External URL to an Internal URL

Mar 20, 2012

Example Inc. wants to hide its actual server room configuration from users to improve security on its Web servers.

To do this, you would create a rewrite action with the values as shown in the following tables. For request headers, the action in the table modifies www.example.com to web.hq.example.net. For response headers, the action does the opposite, translating web.hq.example.net to www.example.com.

Action Name	Type of Rewrite Action	Expression to choose target reference	String expression for replacement text
Action-Rewrite-Request_Server_Replace	REPLACE	HTTP.REQ.HOSTNAME.SERVER	"Web.hq.example.net"
Action-Rewrite-Response_Server_Replace	REPLACE	HTTP.RES.HEADER("Server")	"www.example.com"

Next, you would create rewrite policies using the values shown in the following tables. The first policy checks incoming requests to see if they are valid, and if they are, it performs the Action-Rewrite-Request_Server_Replace action. The second policy checks responses to see if they originate at the server web.hq.example.net. If they do, it performs the Action-Rewrite-Response_Server_Replace action.

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-Request_Server_Replace	Action-Rewrite-Request_Server_Replace	NOREWRITE	HTTP.REQ.HOSTNAME.SERVER.EQ("www.example.com")
Policy-Rewrite-Response_Server_Replace	Action-Rewrite-Response_Server_Replace	NOREWRITE	HTTP.RES.HEADER("Server").EQ("web.hq.example.net")

Finally, you would bind the rewrite policies, assigning each a priority of 500 because they are in different policy banks and therefore will not conflict. You should set the goto expression to NEXT for both bindings.

All instances of www.example.com in the request headers are now changed to web.hq.example.net, and all instances of web.hq.example.net in response headers are now changed to www.example.com.

Example 6: Migrating Apache Rewrite Module Rules

Mar 20, 2012

Example Inc., is currently using the Apache rewrite module to process search requests sent to its Web servers and redirect those requests to the appropriate server on the basis of information in the request URL. Example Inc. wants to simplify its setup by migrating these rules onto the NetScaler platform.

Several Apache rewrite rules that Example currently uses are shown below. These rules redirect search requests to a special results page if they do not have a SiteID string or if they have a SiteID string equal to zero (0), or to the standard results page if these conditions do not apply.

The following are the current Apache rewrite rules:

- RewriteCond %{REQUEST_FILENAME} ^/search$ [NC]
- RewriteCond %{QUERY_STRING} !SiteId= [OR]
- RewriteCond %{QUERY_STRING} SiteId=0
- RewriteCond %{QUERY_STRING} CallName=DisplayResults [NC]
- RewriteRule ^.*$ /results2.html [P,L]
- RewriteCond %{REQUEST_FILENAME} ^/search$ [NC]
- RewriteCond %{QUERY_STRING} CallName=DisplayResults [NC]
- RewriteRule ^.*$ /results.html [P,L]

To implement these Apache rewrite rules on the NetScaler, you would create rewrite actions with the values in the following tables.

Action Name	Type of Rewrite Action	Expression to choose target reference	String expression for replacement text
Action-Rewrite-Display_Results_NulSiteID	REPLACE	HTTP.REQ.URL	"/results2.html"
Action-Rewrite-Display_Results	REPLACE	HTTP.REQ.URL	"/results2.html"

You would then create rewrite policies with the values as shown in the tables below.

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-Display_Results_NulSiteID	Action-Rewrite-Display_Results_NulSiteID	NOREWRITE	HTTP.REQ.URL.PATH.SET_TEXT_MODE(IGNORECASE).EQ("/search") && (!HTTP.REQ.URL.QUERY.CONTAINS("SiteId=") || HTTP.REQ.URL.QUERY.CONTAINS("SiteId=0") || HTTP.REQ.URL.QUERY.SET_TEXT_MODE(IGNORECASE).CONTAINS("CallName=DisplayResults"))
Policy-Rewrite-Display_Results	Action-Rewrite-Display_Results	NOREWRITE	HTTP.REQ.URL.PATH.SET_TEXT_MODE(IGNORECASE).EQ("/search") || HTTP.REQ.URL.QUERY.SET_TEXT_MODE(IGNORECASE).CONTAINS("CallName=DisplayResults"))

Finally, you would bind the rewrite policies, assigning the first a priority of 600 and the second a priority of 700, and then set the goto expression to NEXT for both bindings.

The NetScaler now handles these search requests exactly as the Web server did before the Apache rewrite module rules were migrated.

Example 7: Marketing Keyword Redirection

Mar 20, 2012

The marketing department at Example Inc. wants to set up simplified URLs for certain predefined keyword searches on the company's Web site. For these keywords, it wants to redefine the URL as shown below.

- External URL: http://www.example.com/<marketingkeyword>
- Internal URL: http://www.example.com/go/kwsearch.asp?keyword=<marketingkeyword>

To set up redirection for marketing keywords, you would create a rewrite action with the values in the following table.

Action Name	Type of Rewrite Action	Expression to choose target location	String expression for replacement text
Action-Rewrite-Modify_URL	INSERT_BEFORE	HTTP.REQ.URL.PATH.GET(1)	""go/kwsearch.aspkeyword="|"

You would then create a rewrite policy with the values in the following table.

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-Modify_URL	Action-Rewrite-Modify_URL	NOREWRITE	HTTP.REQ.HOSTNAME.SERVER.EQ("www.example.com")

Finally, you would bind the rewrite policy, assigning it a priority of 800. Unlike the previous rewrite policies, this policy should be the last to be applied to a request that matches its criteria. For this reason, NetScaler administrator sets its Goto Priority Expression to END.

Any request using a marketing keyword is redirected to the keyword search CGI page, whereupon a search is performed and all remaining policies are skipped.

Example 8: Redirect Queries to the Queried Server

Mar 20, 2012

Example Inc. wants to redirect query requests to the appropriate server, as shown here.

- Request: GET /query.cgi?server=5HOST: www.example.com
- Redirect URL: http://web-5.example.com/

To implement this redirection, you would first create a rewrite action with the values in the following table.

Action Name	Type of Rewrite Action	Expression to choose target reference	String expression for replacement text
Action-Rewrite-Replace_Hostheader	REPLACE	HTTP.REQ.HEADER("Host").BEFORE_STR(".example.com")	"server-" + HTTP.REQ.URL.QUERY.VALUE("web")

You would then create a rewrite policy with the values in the following table.

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-Replace_Hostheader	Action-Rewrite-Replace_Hostheader	NOREWRITE	HTTP.REQ.HEADER("Host").EQ("www.example.com")

Finally, you would bind the rewrite policy, assigning it a priority of 900. Because this policy should be the last policy applied to a request that matches its criteria, you set the goto expression to END.

Incoming requests to any URL that begins with http://www.example.com/query.cgi?server= are redirected to the server number in the query.

Example 9: Home Page Redirection

Mar 20, 2012

New Company, Inc. recently acquired a smaller competitor, Purchased Company, and wants to redirect the home page for Purchased Company to a new page on its own Web site, as shown here.

- Old URL: http://www.purchasedcompany.com/*
- New URL: http://www.newcompany.com/products/page.htm

To redirect requests to the Purchased Company home page, you would create rewrite actions with the values in the following table.

Action Name	Type of Rewrite Action	Expression to choose target reference	String expression for replacement text
Action-Rewrite-Replace_URLr	REPLACE	HTTP.REQ.URL.PATH_AND_QUERY	"/products/page.htm"
Action-Rewrite-Replace_Host	REPLACE	HTTP.REQ.HOSTNAME	"www.newcompany.com"

You would then create rewrite policies with the values in the following table.

Policy Name	Action Name	Undefined Action	Expression
Policy-Rewrite-Replace-None	Action-Rewrite-Replace-None	NOREWRITE	!HTTP.REQ.HOSTNAME.SERVER.EQ("www.purchasedcompany.com")
Policy-Rewrite-Replace-Host	Action-Rewrite-Replace_Host	NOREWRITE	HTTP.REQ.HOSTNAME.SERVER.EQ("www.purchasedcompany.com")
Policy-Rewrite-Replace-URL	Action-Rewrite-Replace_URL	NOREWRITE	HTTP.REQ.IS_VALID

Finally, you would bind the rewrite policies globally, assigning the first a priority of 100, the second a priority of 200, and the third a priority of 300. These policies should be the last policies applied to a request that matches the criteria. For this reason, set the goto expression to END for the first and third policies, and to 300 for the second policy. This ensures that all remaining requests are processed correctly.

Requests to the acquired company's old Web site are now redirected to the correct page on the New Company home page.

URL Transformation

Mar 21, 2012

The URL transformation feature provides a method for modifying all URLs in designated requests from an external version seen by outside users to an internal URL seen only by your Web servers and IT staff. You can redirect user requests seamlessly, without exposing your network structure to users. You can also modify complex internal URLs that users may find difficult to remember into simpler, more easily remembered external URLs.

Note: Before you can use the URL transformation feature, you must enable the Rewrite feature. To enable the Rewrite feature, see Enabling the Rewrite Feature.

To begin configuring URL transformation, you create profiles, each describing a specific transformation. Within each profile, you create one or more actions that describe the transformation in detail. Next, you create policies, each of which identifies a type of HTTP request to transform, and you associate each policy with an appropriate profile. Finally, you globally bind each policy to put it into effect.

Configuring URL Transformation Profiles

Oct 29, 2013

A profile describes a specific URL transformation as a series of actions. The profile functions primarily as a container for the actions, determining the order in which the actions are performed. Most transformations transform an external hostname and optional path into a different, internal hostname and path. Most useful transformations are simple and require only a single action, but you can use multiple actions to perform complex transformations.

You cannot create actions and then add them to a profile. You must create the profile first, and then add actions to it. In the CLI, creating an action and configuring the action are separate steps. Creating a profile and configuring the profile are separate steps in both the CLI and the configuration utility.

At the NetScaler command prompt, type the following commands, in the order shown, to create a URL transformation profile and verify the configuration. You can then repeat the second and third commands to configure additional actions:

- add transform profile <profileName> -type URL [-onlyTransformAbsURLinBody (ON | OFF)] [-comment <comment>]
- add transform action <name> <profileName> <priority>
- set transform action <name> [-priority <priority>] [-reqUrlFrom <expression>] [-reqUrlInto <expression>] [-resUrlFrom <expression>] [-resUrlInto <expression>] [-cookieDomainFrom <expression>] [-cookieDomainInto <expression>] [-state (ENABLED | DISABLED)] [-comment "<string>"]
- show transform profile <name>

Example

```
> add transform profile shoppingcart -type URL
 Done
> add transform action actshopping shoppingcart 1000
 Done
> set transform action actshopping -priority 1000 -reqUrlFrom 'shopping.example.com' -reqUrlInto 'www.example.net/shopping' -resUrlFrom 'www.example.net/shopping' -re
 Done
> show transform profile shoppingcart
      Name: shoppingcart
           Type: URL      onlyTransformAbsURLinBody: OFF
      Comment:
      Actions:

1)          Priority 1000   Name: actshopping      ENABLED
 Done
```

At the NetScaler command prompt, type the following commands to modify an existing URL transformation profile or action and verify the configuration:

Note: Use a set transform profile or set transform action command, respectively. The set transform profile command takes the same arguments as does the add transform profile command, and set transform action is the same command that was used for initial configuration.

- set transform action <name> [-priority <priority>] [-reqUrlFrom <expression>] [-reqUrlInto <expression>] [-resUrlFrom <expression>] [-resUrlInto <expression>] [-cookieDomainInto <expression>] [-state (ENABLED | DISABLED)] [-comment "<string>"]
- show transform profile <name>

Example

```
> set transform action actshopping -priority 1000 -reqUrlFrom 'searching.example.net' -reqUrlInto 'www.example.net/searching' -resUrlFrom 'www.example.net/searching' -
 Done
> show transform profile shoppingcart
      Name: shoppingcart
           Type: URL      onlyTransformAbsURLinBody: OFF
      Comment:
      Actions:

1)          Priority 1000   Name: actshopping      ENABLED
 Done
```

First remove all actions associated with that profile by typing the following command once for each action:

- rm transform action <name> After you have removed all actions associated with a profile, remove the profile as shown below.
- rm transform profile <name>

1. In the navigation pane, expand Rewrite, expand URL Transformation, and then click Profiles.
2. In the details pane, click Add.
3. In the Create URL Transformation Profile dialog box, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring URL transformation profiles" as follows (asterisk indicates a required parameter):
   - Name*—name
   - Comment—comment
   - Only transform absolute URLs in response body—onlyTransformAbsURLinBody
4. Click Create, and then click Close. A message appears in the status bar, stating that the Profile has been configured successfully.

1. In the navigation pane, expand Rewrite, expand URL Transformation, and then click Profiles.
2. In the details pane, select the profile you want to configure, and then click Open.
3. In the Configure URL Transformation Profile dialog box, do one of the following.

- To create a new action, click Add.
- To modify an existing action, select the action, and then click Open.

4. Fill in the Create URL Transformation Action or Modify URL Transformation Action dialog box by typing or selecting values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring URL transformation profiles" as follows (asterisk indicates a required parameter):
   - Action Name*—name
   - Comments—comment
   - Priority*—priority
   - Request URL from—reqUrlFrom
   - Request URL into—reqUrlInto
   - Response URL from—resUrlFrom
   - Response URL into—resUrlInto
   - Cookie Domain from—cookieDomainFrom
   - Cookie Domain into—cookieDomainInto
   - Enabled—state
5. Save your changes.
   - If you are creating a new action, click Create, and then Close.
   - If you are modifying an existing action, click OK.

   A message appears in the status bar, stating that the Profile has been configured successfully.
6. Repeat step 3 through step 5 to create or modify any additional actions.
7. To delete an action, select the action, and then click Remove. When prompted, click OK to confirm the deletion.
8. Click OK to save your changes and close the Modify URL Transformation Profile dialog box.
9. To delete a profile, in the details pane select the profile, and then click Remove. When prompted, click OK to confirm the deletion.

Configuring URL Transformation Policies

Oct 29, 2013

After you create a URL transformation profile, you next create a URL transformation policy to select the requests and responses that the NetScaler should transform by using the profile. URL transformation considers each request and the response to it as a single unit, so URL transformation policies are evaluated only when a request is received. If a policy matches, the NetScaler transforms both the request and the response.

Note: The URL transformation and rewrite features cannot both operate on the same HTTP header during request processing. Because of this, if you want to apply a URL transformation to a request, you must make sure that none of the HTTP headers it will modify are manipulated by any rewrite action.

You must create a new policy. On the command line, an existing policy can only be removed. At the NetScaler command prompt, type the following commands to configure a URL transformation policy and verify the configuration:

- add transform policy <name> <rule> <profileName>
- show transform policy <name>

Example

```
> add transform policy polsearch  HTTP.REQ.URL.SUFFIX.EQ("Searching") prosearching
 Done
> show transform policy polsearch
1)     Name: polsearch
       Rule: HTTP.REQ.URL.SUFFIX.EQ("Searching")
       Profile: prosearching
       Priority: 0
       Hits: 0
 Done
```

At the NetScaler command prompt, type the following command to remove a URL transformation policy:

rm transform policy <name>

Example

```
> rm transform policy polsearch
Done
```

1. In the navigation pane, expand Rewrite, expand URL Transformation, and then click Policies.
2. In the details pane, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the policy, and then click Open.
3. In the Create URL Transformation Policy or Configure URL Transformation Policy dialog box, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring URL transformation policies" as follows (asterisk indicates a required parameter):
   - Name*—name (Cannot be changed for a previously configured policy.)
   - Profile*—profileName
   - Expression—rule

If you want help with creating an expression for a new policy, you can either hold down the Control key and press the space bar while your cursor is in the Expression text box. To create the expression, you can type it directly as described below, or you can use the Add Expression dialog box as described in To add an expression by using the Add Expression dialog box.

1. Click Prefix, and choose the prefix for your expression.
   Your choices are:

   - HTTP—The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.
   - SYS—The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   - CLIENT—The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.
   - SERVER—The computer to which the request was sent. Choose this if you want to examine some aspect of the recipient of the request.
   - URL—The URL of the request. Choose this if you want to examine some aspect of the URL to which the request was sent.
   - TEXT—Any text string in the request. Choose this if you want to examine a text string in the request.
   - TARGET—The target of the request. Choose this if you want to examine some aspect of the request target.
     After you choose a prefix, the NetScaler displays a two-part prompt window that displays the possible next choices at the top, and a brief explanation of what the selected choice means at the bottom. The choices depend on which prefix you chose.

2. Select your next term.
   If you chose HTTP as your prefix, your choices are REQ, which specifies HTTP requests, and RES, which specifies HTTP responses. If you chose another prefix, your choices are more varied. For help on a specific choice, click that choice once to display information about it in the lower prompt window.

   When you are certain which choice you want, double-click it to insert it into the Expression window.

3. Type a period, and then continue selecting terms from the list boxes that appear to the right of the previous list box. You type the appropriate text strings or numbers in the text boxes that appear to prompt you to enter a value, until your expression is finished.

4. Click Create or OK, depending on whether you are creating a new policy or modifying an existing policy.
5. Click Close. A message appears in the status bar, stating that the Policy has been configured successfully.

1. In the Create Responder Action or Configure Responder Action dialog box, click Add.
2. In the Add Expression dialog box, in the first list box choose the first term for your expression.
   HTTP

   The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.
   SYS

   The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   CLIENT

The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.

SERVER

The computer to which the request was sent. Choose this if you want to examine some aspect of the recipient of the request.

URL

The URL of the request. Choose this if you want to examine some aspect of the URL to which the request was sent.

TEXT

Any text string in the request. Choose this if you want to examine a text string in the request.

TARGET

The target of the request. Choose this if you want to examine some aspect of the request target.

When you make your choice, the rightmost list box lists appropriate terms for the next part of your expression.

3. In the second list box, choose the second term for your expression. The choices depend upon which choice you made in the previous step, and are appropriate to the context. After you make your second choice, the Help window below the Construct Expression window (which was blank) displays help describing the purpose and use of the term you just chose.
4. Continue choosing terms from the list boxes that appear to the right of the previous list box, or typing strings or numbers in the text boxes that appear to prompt you to enter a value, until your expression is finished.

Globally Binding URL Transformation Policies

Oct 29, 2013

After you have configured your URL transformation policies, you bind them to Global or a bind point to put them into effect. After binding, any a request or response that matches a URL transformation policy is transformed by the profile associated with that policy.

When you bind a policy, you assign a priority to it. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer. In the NetScaler OS, policy priorities work in reverse order - the higher the number, the lower the priority.

Because the URL transformation feature implements only the first policy that a request matches, not any additional policies that it might also match, policy priority is important for achieving the results that you intend. If you give your first policy a low priority (such as 1000), you tell the NetScaler to perform it only if other policies with a higher priority do not match a request. If you give your first policy a high priority (such as 1), you tell the NetScaler to perform it first, and skip any other policies that might also match. You can leave yourself plenty of room to add other policies in any order, without having to reassign priorities, by setting priorities with intervals of 50 or 100 between each policy when you globally bind your policies.

Note: URL transformation policies cannot be bound to TCP-based virtual servers.

At the NetScaler command prompt, type the following commands to globally bind a URL transformation policy and verify the configuration:

- bind transform global <policyName> <priority>
- show transform global

Example

```
> bind transform global polisearching 100
 Done
> show transform global
1)    Policy Name: polisearching
      Priority: 100

 Done
```

1. In the navigation pane, expand Rewrite, then expand URL Transformation, and then click Policies.
2. In the details pane, click Policy Manager.
3. In the Transform Policy Manager dialog box, choose the bind point to which you want to bind the policy. The choices are:
   - **Override Global.** Policies that are bound to this bind point process all traffic from all interfaces on the NetScaler appliance, and are applied before any other policies.
   - **LB Virtual Server.** Policies that are bound to a load balancing virtual server are applied only to traffic that is processed by that load balancing virtual server, and are applied before any Default Global policies. After selecting LB Virtual Server, you must also select the specific load balancing virtual server to which you want to bind this policy.

- **CS Virtual Server.** Policies that are bound to a content switching virtual server are applied only to traffic that is processed by that content switching virtual server, and are applied before any Default Global policies. After selecting CS Virtual Server, you must also select the specific content switching virtual server to which you want to bind this policy.
- **Default Global.** Policies that are bound to this bind point process all traffic from all interfaces on the NetScaler appliance.
- **Policy Label.** Policies that are bound to a policy label process traffic that the policy label routes to them. The policy label controls the order in which policies are applied to this traffic.

4. Select Insert Policy to insert a new row and display a drop-down list with all available, unbound URL transformation policies.
5. Select the policy you want to bind, or select New Policy to create a new policy. The policy that you selected or created is inserted into the list of globally bound URL transformation policies.
6. Make any additional adjustments to the binding.
   - To modify the policy priority, click the field to enable it, and then type a new priority. You can also select Regenerate Priorities to renumber the priorities evenly.
   - To modify the policy expression, double click that field to open the Configure Transform Policy dialog box, where you can edit the policy expression.
   - To set the Goto Expression, double click field in the Goto Expression column heading to display the drop-down list, where you can choose an expression.
   - To set the Invoke option, double click field in the Invoke column heading to display the drop-down list, where you can choose an expression
7. Repeat steps 3 through 6 to add any additional URL transformation policies you want to globally bind.
8. Click OK to save your changes. A message appears in the status bar, stating that the Policy has been configured successfully.

Diameter Support for Rewrite

Apr 09, 2014

The Rewrite feature now supports the Diameter protocol. You can configure Rewrite to modify Diameter requests and response as you would HTTP or TCP requests and responses, allowing you to use Rewrite to manage the flow of Diameter requests and make necessary modifications. For example, if the "Origin-Host" value in a Diameter request is inappropriate, you can use Rewrite to replace it with a value that is acceptable to the Diameter server.

To configure the Rewrite feature to replace the Origin-Host in a diameter request with a different value, at the command prompt, type the following commands:

- add rewrite action <actname> replace "DIAMETER.REQ.AVP(264,\"netscaler.example.net\")"
  For <actname>, substitute a name for your new action. The name can consist of from one to 127 characters in length, and can contain letters, numbers, and the hyphen (-) and underscore (_) symbols. For netscaler.example.net, substitute the Host-Origin that you want to use instead of the original Host-Name.
- add rewrite policy <polname> "diameter.req.avp(264).value.eq(\"host.example.com\")" <actname>
  For <polname>, substitute a name for your new policy. As with <actname>, the name can consist of from one to 127 characters in length, and can contain letters, numbers, and the hyphen (-) and underscore (_) symbols. For host.example.com, substitute the name of the Host-Origin that you want to change. For <actname>, substitute the name of the action that you just created.
- bind lb vserver <vservername> -policyName <polname> -priority <priority> -type REQUEST
  For <vservername>, substitute the name of the load balancing virtual server to which you want to bind the policy. For <polname>, substitute the name of the policy you just created. For <priority>, substitute a priority for the policy.

Example

To create a Rewrite action and policy to modify all Diameter Host-Origins of "host.example.com" to "netscaler.example.net", you could add the following action and policy, and bind the policy as shown.

```
> add rewrite action rw_act_replace_avp replace "diameter.req.avp(264)" "diameter.new.avp(264,\"netscaler.example.net\")"
> add rewrite policy rw_diam_pol "diameter.req.avp(264).value.eq(\"client.realm2.net\")" rw_act_replace_avp
> bind lb vserver vs1 -policyName rw_diam_pol -priority 10 -type REQUEST

Done
```

String Maps

Apr 20, 2015

You can use string maps to perform pattern matching in all NetScaler features that use the default policy syntax. A string map is a NetScaler entity that consists of key-value pairs. The keys and values are strings in either ASCII or UTF-8 format. String comparison uses two new functions, MAP_STRING(<string_map_name>) and IS_STRINGMAP_KEY(<string_map_name>).

A policy configuration that uses string maps performs better than one that does string matching through policy expressions, and you need fewer policies to perform string matching with a large number of key-value pairs. String maps are also intuitive, simple to configure, and result in a smaller configuration.

String maps are similar in structure to pattern sets (a pattern set defines a mapping of index values to strings; a string map defines a mapping of strings to strings) and the configuration commands for string maps (commands such as add, bind, unbind, remove, and show) are syntactically similar to configuration commands for pattern sets. Also, as with index values in a pattern set, each key in a string map must be unique across the map. The following table illustrates a string map called url_string_map, which contains URLs as keys and values.

Table 1. String Map "url_string_map"

Key	Value
/url_1.html	http://www.redirect_url_1.com/url_1.html
/url_2.html	http://www.redirect_url_2.com/url_2.html
/url_3.html	http://www.redirect_url_1.com/url_1.html

The following table describes the two functions that have been introduced to enable string matching with keys in a string map. String matching is always performed with the keys. Additionally, the following functions perform a comparison between the keys in the string map and the complete string that is returned by the expression prefix. The examples in the descriptions refer to the preceding example.

Table 2. String Map Functions

Function	Description
<TEXT>.MAP_STRING(<string_map_name>)	Checks whether the value returned by the expression prefix TEXT matches any of the keys in the string map, and returns the value that corresponds to the key. If no key in the string map matches the value returned by the expression prefix, the function returns a null string. The IGNORECASE and NOIGNORECASE functions can be used for case-insensitive and case-sensitive comparison, respectively. Example 1: HTTP.REQ.URL.MAP_STRING("url_string_map") checks whether the string returned by HTTP.REQ.URL is a key in the string map url_string_map. If the value of HTTP.REQ.URL is /url_1.html, the function returns http://www.redirect_url_1.com/url_1.html. Example 2: HTTP.REQ.URL.SET_TEXT_MODE(IGNORECASE).MAP_STRING("url_string_map") checks whether the string returned by HTTP.REQ.URL is a key in the string map url_string_map. The comparison does not consider case. If the string returned by HTTP.REQ.URL is /URL_1.html, the function returns http://www.redirect_url_1.com/url_1.html. Parameters: string_map_name - The string map.
<TEXT>.IS_STRINGMAP_KEY(<string_map_name>)	Returns TRUE if the string returned by the expression prefix TEXT is a key in the string

Function	Description
	map. The IGNORECASE and NOIGNORECASE functions can be used for case-insensitive and case-sensitive string matching, respectively. **Example 1:** HTTP.REQ.URL.IS_STRINGMAP_KEY("url_string_map") returns TRUE if the value of HTTP.REQ.URL is one of the keys in url_string_map. **Example 2:** HTTP.REQ.URL.SET_TEXT_MODE(IGNORECASE). IS_STRINGMAP_KEY("url_string_map") returns TRUE if the value of HTTP.REQ.URL is one of the keys in url_string_map. In this case, key lookup does not consider case. Therefore, the function returns TRUE even if the value of HTTP.REQ.URL is /URL_3.html. **Parameters:** string_map_name - The string map.

You first create a string map and then bind key-value pairs to it. You can create a string map from the command line interface (CLI) or the configuration utility.

To configure a string map by using the command line interface

At the command prompt, do the following:

1. Create a string map.
   add policy stringmap <name> -comment <string>
2. Bind a key-value pair to the string map.
   bind policy stringmap <name> <key> <value>

   Example:

   > bind policy stringmap url_string_map1 "/url_1.html" "http://www.redirect_url_1.com/url_1.html"

To configure a string map by using the configuration utility

Create a string map and bind the key-value pair to the created entity.

Navigate to **AppExpert** > **String Maps**, click **Add** and specify the relevant details.

The following use case involves a responder policy with a redirect action. In the example below, the first four commands create the string map url_string_map and bind the three key-value pairs used in the earlier example. After creating the map and binding the key-value pairs, you create a responder action (act_url_redirects) that redirects the client to the corresponding URL in the string map or to www.default.com. You also configure a responder policy (pol_url_redirects) that checks whether requested URLs match any of the keys in url_string_map and then performs the configured action. Finally, you bind the responder policy to the content switching virtual server that receives the client requests that are to be evaluated.

```
add stringmap url_string_map

bind stringmap url_string_map /url_1.html http://www.redirect_url_1.com/url_1.html

bind stringmap url_string_map /url_2.html http://www.redirect_url_2.com/url_2.html

bind stringmap url_string_map /url_3.html http://www.redirect_url_1.com/url_1.html

add responder action act_url_redirects redirect 'HTTP.REQ.URL.MAP_STRING("url_string_map") ALT "www.default.com"' -bypassSafetyCheck yes

add responder policy pol_url_redirects TRUE act_url_redirects

bind cs vserver csw_redirect -policyname pol_url_redirects -priority 1 -type request
```

String maps are similar in structure to pattern sets (a pattern set defines a mapping of index values to strings; a string map defines a mapping of strings to strings) and the configuration commands for string maps (commands such as add, bind, unbind, remove, and show) are syntactically similar to configuration

commands for pattern sets. Also, as with index values in a pattern set, each key in a string map must be unique across the map. The following table illustrates a string map called url_string_map, which contains URLs as keys and values.

Table 3. String Map "url_string_map"

Key	Value
/url_1.html	http://www.redirect_url_1.com/url_1.html
/url_2.html	http://www.redirect_url_2.com/url_2.html
/url_3.html	http://www.redirect_url_1.com/url_1.html

The following table describes the two functions that have been introduced to enable string matching with keys in a string map. String matching is always performed with the keys. Additionally, the following functions perform a comparison between the keys in the string map and the complete string that is returned by the expression prefix. The examples in the descriptions refer to the preceding example.

Table 4. String Map Functions

Function	Description
<TEXT>.MAP_STRING(<string_map_name>)	Checks whether the value returned by the expression prefix TEXT matches any of the keys in the string map, and returns the value that corresponds to the key. If no key in the string map matches the value returned by the expression prefix, the function returns a null string. The IGNORECASE and NOIGNORECASE functions can be used for case-insensitive and case-sensitive comparison, respectively. Example 1: HTTP.REQ.URL.MAP_STRING("url_string_map") checks whether the string returned by HTTP.REQ.URL is a key in the string map url_string_map. If the value of HTTP.REQ.URL is /url_1.html, the function returns http://www.redirect_url_1.com/url_1.html. Example 2: HTTP.REQ.URL.SET_TEXT_MODE(IGNORECASE).MAP_STRING("url_string_map") checks whether the string returned by HTTP.REQ.URL is a key in the string map url_string_map. The comparison does not consider case. If the string returned by HTTP.REQ.URL is /URL_1.html, the function returns http://www.redirect_url_1.com/url_1.html. Parameters: string_map_name - The string map.
<TEXT>.IS_STRINGMAP_KEY(<string_map_name>)	Returns TRUE if the string returned by the expression prefix TEXT is a key in the string map. The IGNORECASE and NOIGNORECASE functions can be used for case-insensitive and case-sensitive string matching, respectively. Example 1: HTTP.REQ.URL.IS_STRINGMAP_KEY("url_string_map") returns TRUE if the value of HTTP.REQ.URL is one of the keys in url_string_map. Example 2: HTTP.REQ.URL.SET_TEXT_MODE(IGNORECASE). IS_STRINGMAP_KEY("url_string_map") returns TRUE if the value of HTTP.REQ.URL is one of the keys in url_string_map. In this case, key lookup does not consider case. Therefore, the function returns TRUE even if the value of HTTP.REQ.URL is /URL_3.html. Parameters: string_map_name - The string map.

Function	Description

Updated: 2014-07-30

You first create a string map and then bind key-value pairs to it. You can create a string map from the command line interface (CLI) or the configuration utility.

To configure a string map by using the command line interface

At the command prompt, do the following:

1. Create a string map.
   add policy stringmap <name> -comment <string>
2. Bind a key-value pair to the string map.
   bind policy stringmap <name> <key> <value>

   Example:
   > bind policy stringmap url_string_map1 "/url_1.html" "http://www.redirect_url_1.com/url_1.html"

To configure a string map by using the configuration utility

Create a string map and bind the key-value pair to the created entity.

Navigate to AppExpert > String Maps,click Add and specify the relevant details.

You can use string maps in all features that support the newer default policy syntax. For example, string maps can be used in responder redirects and rewrite actions. You can also reuse a given string map in multiple features.

Example: Responder Policy With a Redirect Action

Updated: 2015-04-20

The following use case involves a responder policy with a redirect action. In the example below, the first four commands create the string map url_string_map and bind the three key-value pairs used in the earlier example. After creating the map and binding the key-value pairs, you create a responder action (act_url_redirects) that redirects the client to the corresponding URL in the string map or to www.default.com. You also configure a responder policy (pol_url_redirects) that checks whether requested URLs match any of the keys in url_string_map and then performs the configured action. Finally, you bind the responder policy to the content switching virtual server that receives the client requests that are to be evaluated.

```
add stringmap url_string_map

bind stringmap url_string_map /url_1.html http://www.redirect_url_1.com/url_1.html

bind stringmap url_string_map /url_2.html http://www.redirect_url_2.com/url_2.html

bind stringmap url_string_map /url_3.html http://www.redirect_url_1.com/url_1.html

add responder action act_url_redirects redirect 'HTTP.REQ.URL.MAP_STRING("url_string_map") ALT "www.default.com"' -bypassSafetyCheck yes

add responder policy pol_url_redirects TRUE act_url_redirects

bind cs vserver csw_redirect -policyname pol_url_redirects -priority 1 -type request
```

Traffic Management

Jun 09, 2014

The following topics cover configuration and installation information for NetScaler traffic management features.

Cache Redirection	Analyzes incoming requests and forwards the requests for already cached data to cache servers. Dynamic HTTP requests and non-cacheable requests are forwarded to the origin servers. Cache redirection is a policy-based feature.
Content Switching	Analyzes client requests and redirects the requests to specific servers on the basis of geographical area, authorization credentials, and device from which the request was initiated.
DataStream	Ensures optimal distribution of traffic from the application and web servers to the database servers. Enables you to segment traffic according to information in the SQL query and on the basis of database names, user names, character sets, and packet size.
Domain Name System	Provides authoritative domain name server (ADNS server) functionality for a domain. The NetScaler appliance functions as a DNS end resolver and forwarder, and also helps in name resolution when fully qualified domain names are not configured.
Firewall Load Balancing	Distributes the traffic across multiple firewalls, providing fault tolerance, increased throughput, and high availability.
Global Server Load Balancing	Enables disaster recovery and ensures continuous availability of applications by protecting against points of failure in a wide area network (WAN).
Link Load Balancing	Load balances outbound traffic across multiple Internet connections to transmit packets seamlessly over the best possible link.
Load Balancing	Distributes user requests for web pages and other protected applications across multiple servers to prevent server overloading and failure. Load balancing also provides fault tolerance.
SSL Offload and Acceleration	Offloads SSL processing from a server to the NetScaler appliance to accelerate SSL transactions.
Web 2.0 Push	Offloads connection management from Web 2.0 servers. Instead of having to maintain a long-lived connection to each client, a Web 2.0 server can maintain a single connection to the NetScaler appliance. The appliance relays the server's data to waiting clients over the connections that it maintains with them.

Cache Redirection

Mar 30, 2012

In a typical deployment, different clients ask web servers for the same content repeatedly. To relieve the origin web server of processing each request, a NetScaler® appliance with cache redirection enabled can serve this content from a cache server instead of from the origin server.

The NetScaler analyzes incoming requests, sends requests for cacheable data to cache servers, and sends non-cacheable requests and dynamic HTTP requests to origin servers.

Cache redirection is a policy-based feature. By default, requests that match a policy are sent to the origin server, and all other requests are sent to a cache server. For testing or maintenance, you might want to skip policy evaluation and direct all requests to the cache or to the origin server.

You can combine content switching with cache redirection to cache selective content and serve content from specific cache servers for specific types of requested content.

A NetScaler configured for cache redirection can be deployed at the edge of a network, in front of the origin server, or anywhere along the network backbone. In an edge deployment, commonly used by Internet Service Providers (ISPs), cable companies, content delivery distribution networks, and enterprise networks, the NetScaler resides directly in front of the clients. In a server-side deployment, the NetScaler is closer to the origin servers.

Cache redirection is used most commonly with the HTTP service type, but it also supports the secure HTTPS protocol.

Cache Redirection Policies

Mar 30, 2012

A cache redirection virtual server applies cache redirection policies to each incoming request. By default, if a request matches one of the configured policies, it is considered non-cacheable, and the NetScaler appliance sends it to the origin server. Other requests are sent to a cache server. This behavior can be reversed, so that requests that match configured cache redirection policies are sent to cache servers.

The NetScaler provides a set of policies for cache redirection. If these built-in policies are not adequate for your deployment, you can configure user-defined cache redirection policies.

Note: Once you have determined which built-in cache redirection policies to use, or have created user-defined policies, proceed with configuring cache redirection. To use this feature, you must configure at least one cache redirection virtual server, and, for normal operation, you must bind at least one cache redirection policy to that virtual server.

Built-in Cache Redirection Policies

May 14, 2015

The NetScaler appliance provides built-in cache redirection policies that handle typical cache requests. These policies are based on HTTP methods, the URL or URL tokens of the incoming request, the HTTP version, or the HTTP headers and their values in the request.

Built-in cache redirection policies can be directly bound to a virtual server and do not need further configuration.

Cache redirection policies use the simpler of two NetScaler expressions languages, called *classic expressions*. For a complete description of classic expressions and how to configure them, see

The NetScaler provides the following built-in cache redirection policies

built in Policy Name	Description
bypass-non-get	Bypass the cache if the request uses an HTTP method other than GET.
bypass-cache-control	Bypass the cache if the request header contains a Cache-Control: no-cache or Cache-Control: no-store header, of if the HTTP request contains a pragma header.
bypass-dynamic-url	Bypass the cache if the URL suggests that the content is dynamic, as indicated by the presence of any of the following extensions: • cgi • asp • exe • cfm • ex • shtml • htx Also bypass the cache if the URL starts with any of the following: • /cgi-bin/ • /bin/ • /exec/
bypass-urltokens	Bypass the cache because the request is dynamic, as indicated by one of the following tokens in the URL: ?, !, or =.
bypass-cookie	Bypass the cache for any URL that has a cookie header and an extension other than .gif or .jpg.

Updated: 2013-08-23

You can display the available cache redirection polices by using the command line interface or the configuration utility.

To display the built-in cache redirection policies by using the command line interface

At the command prompt, type:

show cr policy [<policyName>]
Example

```
> show cr policy
1)      Cache-By-Pass RULE: NS_NON_GET        Policy:bypass-non-get
2)      Cache-By-Pass RULE: (NS_CACHECONTROL_NOSTORE || NS_CACHECONTROL_NOCACHE || NS_HEADER_PRAGMA)    Policy:bypass-cache-control
3)      Cache-By-Pass RULE: (NS_EXT_CGI || NS_EXT_ASP || NS_EXT_EXE || NS_EXT_CFM || NS_EXT_EX || NS_EXT_SHTML || NS_EXT_HTX) || (NS_URL_PATH_CGIBIN || NS_URL_I
4)      Cache-By-Pass RULE: NS_URL_TOKENS      Policy:bypass-urltokens
5)      Cache-By-Pass RULE: (NS_HEADER_COOKIE && NS_EXT_NOT_GIF && NS_EXT_NOT_JPEG)    Policy:bypass-cookie
 Done
>
```

To display the built-in cache redirection policies by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Policies. The configured cache redirection policies appear in the details pane.
2. Select one of the configured policies to view details.

Configuring a Cache Redirection Policy

May 14, 2015

A cache redirection policy includes one or more expressions (also called *rules*). Each expression represents a condition that is evaluated when the client request is compared to the policy.

You do not explicitly configure actions for cache redirection policies. By default, the NetScaler appliance considers any request that matches a policy to be non-cacheable and directs the request to the origin server instead of the cache.

Cache redirection uses the *classic policy* format. Each policy has a name and includes an expression or a set of expressions that are combined by using logical operators.

At the command prompt, type the following commands to add a cache redirection policy and verify the configuration:

- add cr policy <policyName> **-rule** <expression>
- show cr policy [<policyName>]

Examples

Policy with a simple expression:

```
> add cr policy Policy-CRD-1 -rule "REQ.HTTP.URL != /*.jpeg"
 Done
> show cr policy Policy-CRD-1
        Cache-By-Pass RULE: REQ.HTTP.URL != '/*.jpeg'   Policy:Policy-CRD-1
 Done
>
```

Policy with a compound expression:

```
> add cr policy Policy-CRD-2 -rule "REQ.HTTP.METHOD == POST && (REQ.HTTP.URL == /*.cgi || REQ.HTTP.URL != /*.gif)"
 Done
> show cr policy Policy-CRD-2
        Cache-By-Pass RULE: REQ.HTTP.METHOD == POST && (REQ.HTTP.URL == '/*.cgi' || REQ.HTTP.URL != '/*.gif')   Policy:Policy-CRD-2
 Done
>
```

Policy that evaluates a header:

```
> add cr policy Policy-CRD-3 -rule "REQ.HTTP.HEADER If-Modified-Since EXISTS"
 Done
> show cr policy Policy-CRD-3
        Cache-By-Pass RULE: REQ.HTTP.HEADER If-Modified-Since EXISTS    Policy:Policy-CRD-3
 Done
>
```

- To modify a cache redirection policy, use the set cr policy command, which is just like add cr policy command, except that you enter the name of an existing policy.
- To remove a policy, use the rm cr policy command, which accepts only the <name> argument. If the policy is bound to a virtual server, you have to unbind the policy, before you can remove it.

For the details of unbinding a cache redirection policy, see "Unbinding a Policy from a Cache Redirection Virtual Server."

1. Navigate to Traffic Management > Cache Redirection > Policies.
2. In the details pane, click Add.
3. In the Create Cache Redirection Policy dialog box, in the Name* text box, type the name of the policy, and then in the Expression area, click Add.
4. To configure a simple expression, enter the expression. Following is an example of an expression that checks for a .jpeg extension in a URL:

- Expression Type-General
- Flow Type -REQ
- Protocol -HTTP
- Qualifier -URL
- Operator - !=
- Value* - /*.jpeg

The simple expression in the following example checks for an If-Modified-Since header in a request:

- Expression Type -General
- Flow Type -REQ
- Protocol -HTTP
- Qualifier -HEADER
- Operator -EXISTS
- Header Name -If-Modified-Since

5. When you are finished entering the expression, click OK or Create, and then click Close.

1. Navigate to Traffic Management > Cache Redirection > Policies.
2. In the details pane, click Add.
3. In the Name text box, enter a name for the policy.

   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), and underscore (_) symbols. You should choose a name that will make it easy for others to tell what type of content this policy was created to detect.

4. Choose the type of compound expression that you want to create. Your choices are:
   - **Match Any Expression**. The policy matches the traffic if one or more individual expressions match the traffic.
   - **Match All Expressions**. The policy matches the traffic only if every individual expression matches the traffic.
   - **Tabular Expressions**. Switches the Expressions list to a tabular format with three columns. In the rightmost column, you place one of the following operators:
     - The AND [&&] operator, to require that, to match the policy, a request must match both the current expression and the following expression.
     - The OR [||] operator, to require that, to match the policy, a request must match either the current expression or the following expression, or both. Only if the request does not match either expression does it not match the policy.

     You can also group expressions in nested subgroups by selecting an existing expression and clicking one of the following operators:

     - The BEGIN SUBGROUP [+ ( ] operator, which tells the NetScaler appliance to begin a nested subgroup with the selected expression. (To remove this operator from the expression, click -(.)
     - The END SUBGROUP [+)] operator, which tells the NetScaler appliance to end the current nested subgroup with the selected expression. (To remove this operator from the expression, click -) .)
   - **Advanced Free-Form**. Switches off the Expressions Editor entirely and turns the Expressions list into a text area in which you can type a compound expression. This is both the most powerful and the most difficult method of creating a policy expression, and is recommended only for those thoroughly familiar with the NetScaler classic expressions language.
     For more information about creating classic expressions in the Advanced Free-Form text area, see "Configuring Classic Policies and Expressions".

     Caution: If you switch to Advanced Free Form expression editing mode, you cannot switch back to any of the other modes. Do not choose this expression editing mode unless you are sure that you want to use it.
5. If you chose Match Any Expression, Match All Expressions, or Tabular Expressions, click **Add** to display the Add Expression dialog box.

   You should leave the expression type set to **General** for cache redirection policies.

6. In the Flow Type drop-down list, choose a flow type for your expression.

   The flow type determines whether the policy examines incoming or outgoing connections. You have two choices:

   - **REQ**. Configures the NetScaler appliance to examine incoming connections, or requests.
   - **RES**. Configures the appliance to examine outgoing connections, or responses.
7. In the Protocol drop-down list, choose a protocol for your expression.

   The protocol determines the type of information that the policy examines in the request or response. Depending upon whether you chose REQ or RES in the previous drop-down list, either all four or only three of the following choices are available:

   - **HTTP**. Configures the appliance to examine the HTTP header.
   - **SSL**. Configures the appliance to examine the SSL client certificate. Available only if you chose REQ (requests) in the previous drop-down list.
   - **TCP**. Configures the appliance to examine the TCP header.

- **IP**. Configures the appliance to examine the source or destination IP address.

8. Choose a qualifier for your expression from the Qualifier drop-down list.

The contents of the Qualifier drop-down list depend on which protocol you chose. The following table describes the choices available for each protocol.

Table 1. Cache Redirection Policy Qualifiers Available for Each Protocol

Protocol	Qualifier	Definition
HTTP	METHOD	HTTP method used in the request.
	URL	Contents of the URL header.
	URLTOKENS	URL tokens in the HTTP header.
	VERSION	HTTP version of the connection.
	HEADER	Header portion of the HTTP request.
	URLLEN	Length of the contents of the URL header.
	URLQUERY	Query portion of the contents of the URL header.
	URLQUERYLEN	Length of the query portion of the URL header.
SSL	CLIENT.CERT	SSL client certificate as a whole.
	CLIENT.CERT.SUBJECT	Contents of the client certificate subject field.
	CLIENT.CERT.ISSUER	Client certificate issuer.
	CLIENT.CERT.SIGALGO	Signature algorithm used in the client certificate.
	CLIENT.CERT.VERSION	Client certificate version.
	CLIENT.CERT.VALIDFROM	Date from which the client certificate is valid. (The start date.)
	CLIENT.CERT.VALIDTO	Date after which the client certificate is no longer valid. (The end date.)
	CLIENT.CERT.SERIALNUMBER	Client certificate serial number.
	CLIENT.CIPHER.TYPE	Encryption method used in the client certificate.
	CLIENT.CIPHER.BITS	Number of significant bits in the encryption key.
	CLIENT.SSL.VERSION	SSL version of the client certificate.
TCP	SOURCEPORT	Source port of the TCP connection.
	DESTPORT	Destination port of the TCP connection.
	MSS	Maximum segment size (MSS) of the TCP connection.

Protocol	Qualifier	Definition
IP	SOURCEIP	Source IP address of the connection.
	DESTIP	Destination IP address of the connection.

9. Choose the operator for your expression from the Operator drop-down list.

   Your choices depend on the qualifier you chose in the previous step. The complete list of operators that can appear in this drop-down list is:

   - == . Matches the following text string exactly.
   - != . Does not match the following text string.
   - > . Is greater than the following integer.
   - CONTAINS . Contains the following text string.
   - CONTENTS . The contents of the designated header, URL, or URL query.
   - EXISTS . The specified header or query exists.
   - NOTCONTAINS . Does not contain the following text string.
   - NOTEXISTS . The specified header or query does not exist.

   If you want this policy to operate on requests sent to a specific Host, you can leave the default, the equals (==) sign.

10. If the Value text box is visible, type the appropriate string or number into the text box.
    For example, if you want this policy to select requests sent to the host shopping.example.com, you would type that string in the Value text box.

11. If you chose HEADER as the qualifier, type the header you want in the Header Name text box.
12. Click OK to add your expression to the Expression list.
13. Repeat steps 4 through 11 to create additional expressions.
14. Click Close to close the Add Expression dialog box and return to the Create Cache Redirection Policy dialog box.

Cache Redirection Configurations

Mar 30, 2012

Depending on your deployment and network topology, you can configure one of the following types of cache redirection:

- **Transparent**. A transparent cache can reside on a variety of points along a network backbone to alleviate traffic along the delivery route. In transparent mode, the cache redirection virtual server intercepts all traffic flowing to the NetScaler appliance and applies cache redirection policies to determine whether content should be served from the cache or from the origin server.
- **Forward proxy**. A forward proxy cache server resides on the edge of an enterprise LAN and faces the WAN. In the forward proxy mode, the cache redirection virtual server resolves the hostname of the incoming request by using a DNS server and forwards requests for non-cacheable content to the resolved origin servers. Cacheable requests are sent to the configured cache servers.
- **Reverse proxy**. Reverse proxy caches are configured for specific origin servers. Incoming traffic directed to the reverse proxy, can either be served from a cache server or be sent to the origin server with or without modification to the URL.

Configuring Transparent Redirection

Sep 11, 2013

When you configure transparent cache redirection, the NetScaler appliance evaluates all traffic it receives, to determine whether it is cacheable. This mode alleviates traffic along the delivery route and is often used when the cache server resides on the backbone of an ISP or carrier.

By default, cacheable requests are sent to a cache server, and non-cacheable requests to the origin server. For example, when the NetScaler appliance receives a request that is directed to a web server, it compares the HTTP headers in the request with a set of policy expressions. If the request does not match the policy, the appliance forwards the request to a cache server. If the response does match a policy, the appliance forwards the request, unchanged, to the web server.

For details on how to modify this default behavior, see "Directing Policy Hits to the Cache instead of the Origin."

To configure transparent redirection, first enable cache redirection and load balancing, and configure edge mode. Then, create a cache redirection virtual server with a wildcard IP address (*), so that this virtual server can receive traffic coming to the NetScaler on any IP address the appliance owns. To this virtual server, bind cache redirection policies that describe the types of requests that should not be cached. Then, create a load balancing virtual server that will receive traffic from the cache redirection virtual server for cacheable requests. Finally, create a service that represents a physical cache server and bind it to the load balancing virtual server.

Enabling Cache Redirection and Load Balancing

Oct 31, 2013

The NetScaler cache redirection and load balancing features are not enabled by default. They must be enabled before any cache redirection configuration can take effect.

At the command prompt, type the following command to enable cache redirection and load balancing and verify the settings:

- enable ns feature cr lb
- show ns feature

Example

```
> enable ns feature cr lb
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection          SP              ON
 3)    Load Balancing           LB              ON
 4)    Content Switching         CS              ON
 5)    Cache Redirection         CR              ON
 6)    Sure Connect

         ...
         ...
         ...

 23)   HTML Injection           HTMLInjection      ON
 24)   NetScaler Push           push            OF
 Done
>
```

1. In the navigation pane, expand System, and then click Settings.
2. To enable cache redirection, in the details pane, under Modes and Features, click Configure advanced features.
   1. In Configure Advanced Features dialog box, select the check box next to the Cache Redirection, and then click OK.
   2. In Enable/Disable Feature(s)? dialog box, click Yes.
3. To enable load balancing, in the details pane, under Modes and Features, click Configure basic features.
   1. In Configure Basic Features dialog box, select the check box next to the Load Balancing, and then click OK.
   2. In Enable/Disable Feature(s)? dialog box, click Yes.

Configuring Edge Mode

Oct 31, 2013

When deployed at the edge of a network, the NetScaler appliance dynamically learns about the servers on that network. Edge mode enables the appliance to dynamically learn about up to 40,000 HTTP servers and proxy TCP connections for these servers.

This mode turns off collection of statistics for the dynamically learned services and is typically used in transparent deployments for cache redirection.

At the command prompt, type the following commands to enable edge mode and verify the setting:

- enable ns mode Edge
- show ns mode

Example

```
> enable ns mode edge
 Done

> show ns mode

      Mode                  Acronym          Status
      -------               -------          ------
        ...
        ...
        ...
 6)   MAC-based forwarding       MBF             ON
 7)   Edge configuration         Edge            ON
 8)   Use Subnet IP              USNIP           OFF
        ...
        ...
        ...
 16)  Bridge BPDUs               BridgeBPDUs     OFF
 Done
>
```

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Configure modes.
3. In Configure Modes dialog box, select the check box next to the Edge Configuration, and then click OK.
4. In Enable/Disable Feature(s)? dialog box, click Yes.

Configuring a Cache Redirection Virtual Server

Sep 10, 2013

By default, a cache redirection virtual server forwards cacheable requests to the load balancing virtual server for the cache, and forwards non-cacheable requests to the origin server (except in a reverse proxy configuration, in which non-cacheable requests are sent to a load balancing virtual server). There are three types of cache redirection virtual servers: transparent, forward proxy, and reverse proxy.

A transparent cache redirection virtual server uses an IP address of * and a port number, usually 80, that can accept HTTP traffic sent to any IP address that the NetScaler represents. As a result, you can configure only one transparent cache redirection virtual server. Any additional cache redirection virtual servers that you configure must be forward proxy or reverse proxy redirection servers.

At the command prompt, type the following commands to add a cache redirection virtual server and verify the configuration:

- add cr vserver <name> <serviceType> [<IPAddress> <port>] [-cacheType <cacheType>] [-redirect <redirect>]
- show cr vserver [<name>]

Example

```
add cr vserver Vserver-CRD-1 HTTP * 80 -cacheType TRANSPARENT -redirect POLICY
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: RULE       Cache: TRANSPARENT
      On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF
 Done
>
```

- To modify a virtual server, use the set cr vserver command, which is just like using the add cr vserver command, except that you enter the name of an existing virtual server.
- To remove a virtual server, use the rm cr vserver command, which accepts only the <name> argument.

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Cache Redirection) dialog box, specify values for the following parameters as shown:

- Name*—name
- Port*—port

* A required parameter

4. In the Protocol drop-down list, select a supported protocol (for example, **HTTP**). If the virtual server is to receive traffic on a port other than the standard port for the selected protocol, enter a new value in the Port field.
5. Click the Advanced tab.
6. Verify that Cache Type is set to TRANSPARENT and Redirect is set to POLICY.
7. Click Create, and then click Close. The Cache Redirection Virtual Servers pane displays the new virtual server.
8. Select the new cache redirection virtual server to display the details of its configuration.

Binding Policies to the Cache Redirection Virtual Server

Aug 22, 2013

Cache redirection policies are not automatically bound to the cache redirection virtual server. A policy based cache redirection virtual server cannot function unless you bind at least one policy to it.

At the command prompt, type:

- bind cr vserver <name> -policyName <string>
- show cr vserver [<name>]

Example

```
> bind cr vserver Vserver-CRD-1 -policyName bypass-cache-control
 Done
> bind cr vserver Vserver-CRD-1 -policyName bypass-dynamic-url
 Done
> bind cr vserver Vserver-CRD-1 -policyName bypass-urltokens
 Done
> bind cr vserver Vserver-CRD-1 -policyName bypass-cookie
 Done

> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: RULE       Cache: TRANSPARENT
      On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: bypass-dynamic-url
3)     Cache bypass  Policy: bypass-urltokens
4)     Cache bypass  Policy: bypass-cookie
 Done
>
```

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. Click the virtual server that you want to configure, and click Open.
3. On the Policies tab, select type of the policy and then click Insert Policy.

4. Under Policy Name column, select the policy that you want to bind.
5. Click OK.

Unbinding a Policy from a Cache Redirection Virtual Server

Aug 23, 2013

When you unbind a policy from the cache redirection virtual server, the NetScaler appliance no longer applies the policy when evaluating client requests.

At the command prompt, type:

- unbind cr vserver <name> -policyName <string>
- show cr vserver [<name>]

Example

```
unbind cr vserver Vserver-CR-1 -policyName bypass-non-get
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: RULE       Cache: TRANSPARENT
      On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON      Via: ON ARP: OFF

1)     Cache bypass  Policy: bypass-cache-control
 Done
>
```

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. Click the virtual server that you want to configure, and then click Open.
3. On the Policies tab, under Policy Name, select the policy that you want to unbind.
4. Click Unbind Policy, and then click OK.

Creating a Load Balancing Virtual Server

Sep 10, 2013

The cache redirection virtual server on the NetScaler appliance can send requests to either a cache server farm, if the request is cacheable, or to the origin server farm if the request is not cacheable.

Each cache server is represented on the appliance by a service, which is bound to a load balancing virtual server that receives requests from the cache redirection virtual server and forwards those requests to the servers.

For details on configuring load balancing virtual servers and other configuration options, see "Load Balancing."

At the command prompt, type the following commands to create a load balancing virtual server and verify the configuration:

- add lb vserver <name> <serviceType> [<IPAddress>] [<port>]
- show lb vserver [<name>]

Example

```
> add lb vserver Vserver-LB-CR HTTP 10.102.20.30 80
 Done
> show lb vserver Vserver-LB-CR
        Vserver-LB-CR (10.102.20.30:80) - HTTP  Type: ADDRESS
        State: DOWN
        Last state change was at Fri Jul  2 08:47:52 2010
        Time since last state change: 0 days, 00:00:08.470
        Effective State: DOWN
        Client Idle Timeout: 180 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Port Rewrite : DISABLED
        No. of Bound Services :  0 (Total)      0 (Active)
        Configured Method: LEASTCONNECTION
        Mode: IP
        Persistence: NONE
        Vserver IP and Port insertion: OFF
        Push: DISABLED  Push VServer:
        Push Multi Clients: NO
        Push Label Rule: none
 Done
>
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters as shown:

- Name*-name
- IP Address*- IPAddress
- Port*-port

* A required parameter

4. In the Protocol* drop down list, select a supported protocol (for example, **HTTP**). If the virtual server is to receive traffic on a port other than the well-known port for the selected protocol, enter a new value in the Port field.
5. Click Create, and then click Close. The Load Balancing Virtual Servers pane displays the new virtual server.

Configuring an HTTP Service

Sep 10, 2013

On the NetScaler appliance, a service represents a physical server on the network. In the transparent cache redirection configuration, the service represents the cache server. Cacheable requests are sent by the cache redirection virtual server to the load balancing virtual server, which in turn forwards each request to the correct service, which passes it on to the cache server.

At the command prompt, type the following commands to create an HTTP service and verify the configuration:

- add service <name> <IP> <serviceType> <port> -cacheType <cacheType>
- show service [<name>]

Example

```
> add service Service-HTTP-1 10.102.29.40 HTTP 80 -cacheType TRANSPARENT
 Done
> show service Service-HTTP-1
      Service-HTTP-1 (10.102.29.40:80) - HTTP
      State: DOWN
      Last state change was at Fri Jul  2 09:14:17 2010
      Time since last state change: 0 days, 00:00:13.820
      Server Name: 10.102.29.40
      Server ID : 0   Monitor Threshold : 0
      Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
      Use Source IP: NO
      Client Keepalive(CKA): NO
      Access Down Service: NO
      TCP Buffering(TCPB): NO
      HTTP Compression(CMP): YES
      Idle timeout: Client: 180 sec   Server: 360 sec
      Client IP: DISABLED
      Cache Type: TRANSPARENT Redirect Mode:
      Cacheable: NO
      SC: OFF
      SP: ON
      Down state flush: ENABLED

1)     Monitor Name: tcp-default
            State: DOWN     Weight: 1
            Probes: 3       Failed [Total: 3 Current: 3]
            Last response: Failure - Time out during TCP connection establishment stage
            Response Time: N/A
 Done
>
```

- To modify a service, use the set service command, which is just like using the add service command, except that you enter the name of an existing service.
- To remove a service, use the rm service command, which accepts only the <name> argument.

1. Navigate to Traffic Management > Load Balancing > Services
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:

   - Service Name*—name
   - Server*— IP
   - Port*—port

   * A required parameter

4. In the Protocol* drop-down list, select a supported protocol (for example, **HTTP**).
5. Click Create, and then click Close.

Binding/Unbinding a Service to/from a Load Balancing Virtual Server

Aug 22, 2013

You must bind a service to the load balancing virtual server. This enables the load balancer to forward the request to the server that the service represents. If your configuration changes, you can unbind a service from the load balancing virtual server.

To bind a service to a load balancing virtual server by using the command line interface

At the command prompt, type:

- bind lb vserver <name> <serviceName>
- show lb vserver [<name>]

Example

```
> bind lb vserver vserver-LB-CR service-HTTP-1
 Done
> show lb vserver Vserver-LB-CR
      Vserver-LB-CR (10.102.20.30:80) - HTTP  Type: ADDRESS
      State: DOWN
      Last state change was at Fri Jul  2 08:47:52 2010
      Time since last state change: 0 days, 00:42:25.610
      Effective State: DOWN
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Port Rewrite : DISABLED
      No. of Bound Services :  1 (Total)      0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
      Vserver IP and Port insertion: OFF
      Push: DISABLED  Push VServer:
      Push Multi Clients: NO
      Push Label Rule: none

1) Service-HTTP-1 (10.102.29.40: 80) - HTTP State: DOWN Weight: 1
 Done
>
```

To unbind a service from a load balancing virtual server by using the command line interface

To unbind a service, use the unbind lb vserver command instead of bind lb vserver.

To bind/unbind a service from a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers
2. In the details pane, select the virtual server from which you want to bind/unbind the service, and then click Open.

3. On the Services tab, in the Active column, select/clear the check box next to the Service Name.
4. Click OK.

Disabling the Use the Proxy Port Setting for Transparent Caching

Sep 11, 2013

If the use source IP (USIP) option is disabled on a cache service configured on the NetScaler appliance, the appliance forwards client requests to the cache service by using a NetScaler-owned subnet IP (SNIP) address or mapped IP (MIP) address as the source IP address and a random port as the source port. The randomly selected port is called the proxy port.

However, if you want to configure a fully transparent cache (a cache configuration in which the cache service receives the client's IP address and port number), you must not only enable the USIP option, either globally or on the cache service, but also disable the Use Proxy Port setting, either globally or on the cache service. Disabling the Use Proxy Port setting enables the appliance to use the client's source port as the source port when it connects to the cache service, and ensures a fully transparent cache configuration.

For more information about configuring the Use Proxy Port option globally or on a service, see "Configuring the Source Port for Server-Side Connections."

Assigning a Port Range to the NetScaler

Aug 22, 2013

Sharing of the client IP address may create a conflict that makes network devices, such as routers, cache servers, origin servers, and other NetScaler appliances, unable to determine the appliance, and therefore the client, to which the response should be sent.

A method to solve this problem is to assign a source port range to the NetScaler appliance. This allotment enables network devices to unambiguously identify the NetScaler appliance that sent the request.

To assign a source port range to a NetScaler appliance by using the command line interface

At the command prompt, type:

set ns param -crPortRange <startPortNumber-endPortNumber>

To assign a source port range to a NetScaler appliance by using the NetScaler configuration utility

1. In the navigation pane, click System, and then click Settings.
2. In the Settings group, click the Change global system settings link.
3. In the Cache Redirection Port Range group, specify the port range for the NetScaler by typing a port number for Start Port and a port number for End Port.
4. Click OK.

Enabling Load Balancing Virtual Servers to Redirect Requests to Cache

Aug 23, 2013

If a load balancing virtual server is configured to listen on a particular IP address and port combination, it takes precedence over the cache redirection virtual server for any requests destined for that address-port combination. Therefore, the cache redirection virtual server does not process those requests.

If you want to override this functionality and let the cache redirection virtual server decide whether the request should be served from the cache or not, configure the particular load balancing virtual server to be cacheable.

Such a configuration is typically used when an ISP uses a NetScaler appliance at the edge of its network and all traffic flows through the appliance.

To enable load balancing virtual servers to redirect requests to the cache by using the command line interface

At the command prompt, type:

- set lb vserver <name> [-cacheable (YES | NO)]
- show lb vserver [<name>]

Example

```
set lb vserver Vserver-LB-CR –cacheable YES
> show lb vserver vserver-LB-CR
      Vserver-LB-CR (10.102.20.30:80) - HTTP  Type: ADDRESS
      State: DOWN
      Last state change was at Fri Jul  2 08:47:52 2010
      Time since last state change: 0 days, 01:05:51.510
      Effective State: DOWN
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Port Rewrite : DISABLED
      No. of Bound Services :  1 (Total)      0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
      Cacheable: YES  PQ: OFF SC: OFF
      Vserver IP and Port insertion: OFF
      Push: DISABLED  Push VServer:
      Push Multi Clients: NO
      Push Label Rule: none

1) Service-HTTP-1 (10.102.29.40: 80) - HTTP State: DOWN Weight: 1
 Done
```

For transparent cache redirection, the NetScaler intercepts all traffic and evaluates every request to determine whether it is cacheable. Non-cacheable requests are sent unchanged to the origin server.

When using transparent cache redirection, you may want to turn off cache redirection for load balancing virtual servers that always direct traffic to origin servers.

To turn off caching for a load balancing virtual server by using the command line interface

To turn off caching for a load balancing virtual, use the unset lb vserver command instead of set lb vserver. Specify a value of NO value for the **-cacheable** parameter**.**

To enable or disable load balancing virtual servers to redirect requests to the cache by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server from which you want to enable/disable the caching, and then click Open.
3. On the Advanced tab, select/clear Cache Redirection check box.
4. Click OK.

Configuring Forward Proxy Redirection

Dec 17, 2013

A forward proxy is a single point of contact for a client or group of clients. In this configuration, the NetScaler appliance redirects non-cacheable requests to an origin server and redirects cacheable requests to either a forward proxy cache or a transparent cache.

When the NetScaler is configured as a forward proxy, users must modify their browsers so that the browser sends requests to the forward proxy instead of the destination servers.

A forward proxy cache redirection virtual server on the NetScaler compares the request with a policy for caching. If the request is not cacheable, the NetScaler queries a DNS load balancing virtual server for resolution of the destination, and then sends the request to the origin server. If the request is cacheable, the NetScaler forwards the request to a load balancing virtual server for the cache.

The NetScaler relies on a host domain name or IP address in the request's HOST header to determine the requested destination. If there is no HOST header in the request, the appliance inserts a HOST header based on the destination IP address in the request.

Typically, the NetScaler appliance acts as a forward proxy in an enterprise LAN. In such a configuration, the appliance resides at the edge of an enterprise LAN and intercepts client requests before they are fanned out to the WAN. Configuring the appliance in the forward proxy mode reduces traffic on the WAN.

To configure forward proxy cache redirection, first enable load balancing and cache redirection on the NetScaler. Then, configure a DNS load balancing virtual server and associated services. Also configure a load balancing virtual server and bind to it appropriate services for the cache. Configure a forward proxy cache redirection virtual server and bind the DNS and load balancing virtual servers to it. You must also configure caching policies and bind them to the cache redirection virtual server. To complete the setup, configure the client browsers to use the forward proxy.

For details on how to enable cache redirection and load balancing on the NetScaler, see "Enabling Cache Redirection and Load Balancing."

For details on how to create a load balancing virtual server, see "Creating a Load Balancing Virtual Server."

For details on how to configure services that represent the cache server, see "Configuring an HTTP Service."

For details on how to bind the service to a virtual server, see "Binding Services to the Virtual Server."

For details on how to create a forward proxy cache redirection server, see "Configuring a Cache Redirection Virtual Server", and create a virtual server of type TRANSPARENT or FORWARD.

For details on binding cache redirection policies to the cache redirection virtual server, see "Configuring a Cache Redirection Policy."

Creating a DNS Service

Sep 10, 2013

A DNS service is a representation, on the NetScaler appliance, of a physical DNS server in the network. A DNS load balancing virtual server sends DNS requests to the DNS server in the network through such a service.

To create a DNS service by using the command line interface

At the command line, type the following commands to create a DNS service and verify the configuration :

- add service <name> <IP> <serviceType> <port>
- show service [<name>]

Example

```
add service Service-DNS-1 10.102.29.41 DNS 53
show service Service-DNS-1
 Service-DNS-1 (10.102.29.41:53) - DNS
      State: DOWN
      Last state change was at Fri Jul  2 10:14:32 2010
      Time since last state change: 0 days, 00:00:13.550
      Server Name: 10.102.29.41
      Server ID : 0   Monitor Threshold : 0
      Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
      Use Source IP: NO
      Client Keepalive(CKA): NO
      Access Down Service: NO
      TCP Buffering(TCPB): NO
      HTTP Compression(CMP): NO
      Idle timeout: Client: 120 sec   Server: 120 sec
      Client IP: DISABLED
      Cacheable: NO
      SC: OFF
      SP: OFF
      Down state flush: ENABLED

1)     Monitor Name: ping-default
            State: DOWN     Weight: 1
            Probes: 3       Failed [Total: 3 Current: 3]
            Last response: Failure - Probe timed out.
            Response Time: 2000.0 millisec
 Done
```

To add an DNS service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:

   - Service Name*—name

- Server*—IP
- Port*—port

* A required parameter

4. In the Protocol* drop down list, select a supported protocol (for example, **DNS**).
5. Click Create, and then click Close.

Creating a DNS Load Balancing Virtual Server

Aug 23, 2013

The DNS virtual server enables the forward proxy to perform DNS resolution before forwarding a client request to an origin server. The DNS load balancing virtual server is associated with the DNS service that represents the physical DNS server on the network.

To create a DNS load balancing virtual server by using the command line interface

At the command line, type the following commands to create a DNS load balancing virtual server and verify the configuration:

- add lb vserver <name> <serviceType>
- show lb vserver [<name>]

Example

```
> add lb vserver Vserver-DNS-1 DNS
 Done
> show lb vserver Vserver-DNS-1
      Vserver-DNS-1 (0.0.0.0:0) - DNS Type: ADDRESS
      State: DOWN
      Last state change was at Fri Jul  2 10:32:28 2010
      Time since last state change: 0 days, 00:00:08.10
      Effective State: DOWN  ARP:DISABLED
      Client Idle Timeout: 120 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      No. of Bound Services :  0 (Total)       0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
 Done
>
```

To create a DNS load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, in the Name box, type a name for the virtual server.
4. In the Protocol* drop down list, select a supported protocol (for example, **DNS**).
5. Click Create, and then click Close. The DNS Virtual Servers pane displays the new virtual server.

Binding the DNS Service to the Virtual Server

Aug 22, 2013

For the DNS server to respond to DNS requests, the service representing the DNS server must be bound to the DNS virtual server.

To bind the DNS service to the load balancing virtual server:

At the command prompt, type the following commands to bind the DNS service to the load balancing virtual server and verify the configuration:

- bind lb vserver <name> <serviceName>
- show lb vserver <name>

Example

```
> bind lb vserver Vserver-DNS-1 Service-DNS-1
 Done
> show lb vserver Vserver-DNS-1
      Vserver-DNS-1 (0.0.0.0:0) - DNS Type: ADDRESS
      State: DOWN
      Last state change was at Fri Jul  2 10:32:28 2010
      Time since last state change: 0 days, 00:12:16.80
      Effective State: DOWN  ARP:DISABLED
      Client Idle Timeout: 120 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      No. of Bound Services :  1 (Total)      0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE

1) Service-DNS-1 (10.102.29.41: 53) - DNS State: DOWN   Weight: 1
 Done
>
```

To unbind a DNS service from the load balancing virtual server:

Use the unbind lb vserver command instead of bind lb vserver.

To Bind/Unbind a DNS service to/from a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers
2. In the details pane, select the virtual server to/from which you want to bind/unbind the DNS service, and then click Open.
3. On the Services tab, in the Active column, select/clear the check box next to the Service Name.
4. Click OK.

Configuring a Client Web Browser to Use a Forward Proxy

Mar 30, 2012

When you configure the NetScaler appliance as forward proxy cache redirection virtual server in the network, you must configure the client Web browser to send requests to the forward proxy. Typically, when you use a forward proxy, the only route to the servers in the network is through the forward proxy.

Refer the documentation for your browser to configure the browser to use a forward proxy. Specify the IP address and port number of the forward proxy cache redirection virtual server for this configuration.

Configuring Reverse Proxy Redirection

Dec 17, 2013

A reverse proxy resides in front of one or more Web servers and shields the origin server from client requests. Often, a reverse proxy cache is a front-end for all client requests to a server. An administrator assigns a reverse proxy cache to a specific origin server. This is unlike transparent and forward proxy caches, which cache frequently requested content for all requests to any origin server, and the choice of a server is based on the request.

Unlike a transparent proxy cache, the reverse proxy cache has its own IP address and can replace destination domains and URLs in a non-cacheable request with new destination domains and URLs.

You can deploy reverse proxy cache redirection at the origin-server side or at the edge of a network. When deployed at the origin server, the reverse proxy cache redirection virtual server is a front-end for all requests to the origin server.

In the reverse proxy mode, when the NetScaler receives a request, a cache redirection virtual server evaluates the request and forwards it to either a load balancing virtual server for the cache or a load balancing virtual server for the origin. The incoming request can be transformed by changing the host header or the host URL before they it is sent to the backend server.

To configure reverse proxy cache redirection, first enable cache redirection and load balancing. Then, configure a load balancing virtual server and services to send cacheable requests to the cache servers. Also configure a load balancing virtual server and associated services for the origin servers. Then, configure a reverse proxy cache redirection virtual server and bind relevant cache redirection policies to it. Finally, configure mapping policies and bind them to the reverse proxy cache redirection virtual server.

The mapping policies have an associated action that enables the cache redirection virtual server to forward any non-cacheable request to the load balancing virtual server for the origin.

Be sure to create the default cache server destination.

For details on how to enable cache redirection and load balancing on the NetScaler, see "Enabling Cache Redirection and Load Balancing."

For details on how to create a load balancing virtual server, see "Creating a Load Balancing Virtual Server."

For details on how to configure services that represent the cache server, see "Configuring an HTTP Service."

For details on how to bind the service to a virtual server, see "Binding Services to the Virtual Server."

For details on how to create a reverse proxy cache redirection server, see "Configuring a Cache Redirection Virtual Server", and create a virtual server of type REVERSE.

For details on binding built-in cache redirection policies to the cache redirection virtual server, see "Binding Policies to the Cache Redirection Virtual Server."

Configuring Mapping Policies

If an incoming request is non-cacheable, the reverse-proxy cache redirection virtual server replaces the domain and URL in the request with the domain and URL of a target origin server and forwards the request to the load balancing virtual server for the origin.

A mapping policy enables the reverse proxy cache redirection virtual server to replace the destination domain and URL and forward the request to the load balancing virtual server for the origin.

A mapping policy must first translate the domain and the URL, and then pass the request on to the origin load balancing virtual server.

A mapping policy can map a domain, a URL prefix, and a URL suffix, as follows:

- Domain mapping: You can map a domain without a prefix or suffix. The domain mapping is the default mapping for the virtual server (for example, mapping www.mycompany.com to www.myrealcompany.com).
- Prefix mapping: You can replace a specified pattern prefixed as part of the URL (for example, mapping www.mycompany.com/sports/index.html to www.mycompany.com/news/index.html).
- Suffix mapping: You can replace the file suffix in the URL (for example, mapping www.mycompany.com/sports/index.html to www.mycompany.com/sports/index.asp).

The source and the destination strings being mapped must be similar. If you specify a source domain, you must specify a destination domain, and if you specify a source suffix, you must specify a destination suffix. Similarly, if you specify an exact URL from the source, the target URL must also be an

exact URL.

Once you configure mapping policies for the reverse proxy mode, you must bind them to the cache redirection virtual server.

You can use combinations of the source URL, target URL, and source and target domains to configure all three types of domain mapping.

To configure a mapping policy for reverse proxy mode by using the command line interface

At the command prompt, type the following command to add a policy map and verify the configuration:

- add policy map <mapPolicyName> -sd <string> [-su <string>] [-td <string>] [-tu <string>]
- show policy map [<mapPolicyName>]

Example

The following command maps a domain in a client request to a target domain:

```
> add policy map myMappingPolicy -sd www.mycompany.com -td www.myrealcompany.com
 Done
> show policy map myMappingPolicy
1)     Name: myMappingPolicy
       Source Domain: www.mycompany.com       Source Url:
       Target Domain: www.myrealcompany.com    Target Url:
 Done
>
```

Following is an example of mapping a URL suffix to a different URL suffix:

```
> add policy map myOtherMappingPolicy -sd www.mycompany.com -td www.myrealcompany.com -su /news.html -tu /realnews.html
 Done
> show policy map myOtherMappingPolicy
1)     Name: myOtherMappingPolicy
       Source Domain: www.mycompany.com       Source Url: /news.html
       Target Domain: www.myrealcompany.com    Target Url: /realnews.html
 Done
>
```

To configure a mapping policy for reverse proxy mode by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Map Policies.
2. In the details pane, click Add.
3. In the Create Map Policy dialog box, specify values for the following parameters as shown:

   - Name*- mapPolicyName
   - Source Domain*-sd
   - Target Domain*-td
   - Source URL-su
   - Target URL-tu

   * A required parameter

4. Click Create, and then click Close. The Map pane displays the new mapping policy.

To bind the mapping policy to the cache redirection virtual server by using the command line interface

At the command prompt, type the following commands to bind the mapping policy to the cache redirection virtual server and verify the configuration:

- bind cr vserver <name> -policyName <string> [<targetVserver>]
- show cr vserver <name>

Example


```
> bind cr vserver Vserver-CRD-3 -policyName myMappingPolicy Vserver-LB-CR
 Done
> show cr vserver Vserver-CRD-3
        Vserver-CRD-3 (10.102.29.50:88) - HTTP  Type: CONTENT
        State: UP
        Client Idle Timeout: 180 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Default: Vserver-LB-CR  Content Precedence: RULE        Cache: REVERSE
        On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
        Redirect: POLICY        Reuse: ON       Via: ON ARP: OFF

1)      Policy:         Target: Vserver-LB-CR   Priority: 0     Hits: 0
1)      Map: myMappingPolicy Target: Vserver-LB-CR
 Done
>
```

To bind the mapping policy to the cache redirection virtual server by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, select the virtual server from which you want to bind the mapping policy, and then click Open.
3. In the Configure Virtual Server(Cache Redirection), on the Policies tab, select Map, and then click Insert Policy.
4. In the Policy Name column, select the policy from drop down list.
5. In the Target column, click the down arrow, and then select the vserver from drop down list.
6. Click OK.

Selective Cache Redirection

Sep 11, 2013

Selective cache redirection sends requests for particular types of content, for example, images, to one cache server or group of cache servers and sends other types of content to a different cache server or group of cache servers. You can configure advanced cache redirection in transparent, reverse proxy, or forward proxy modes.

In selective cache redirection, the NetScaler appliance intercepts a client request and forwards non-cacheable requests to the original destination in the client request. For cacheable requests, the appliance sends the requests to the destination cache server that can serve content of a specific content type.

Selective cache redirection involves configuring content switching policies in addition to cache redirection policies. The NetScaler first evaluates the cache redirection policies that are bound to the cache redirection virtual server. If a request matches a cache redirection policy, the cache redirection virtual server sends the request to the origin server or a load balancing virtual server for the origin. If no cache redirection policies match the request, the NetScaler evaluates the content switching policies bound to the cache redirection virtual server. If a content switching policy matches the request, the cache redirection virtual server redirects the request to a load balancing virtual server for the cache.

To configure selective cache redirection, first enable cache redirection, load balancing, and content switching on the NetScaler appliance. Then, configure a load balancing virtual server for the cache and an associated HTTP service. After this, configure a cache redirection virtual server and bind both the cache redirection and content switching policies to it. Once you have bound the policies, you can configure the virtual server to give precedence to either rule based or URL based content-switching policies.

When configured for transparent mode cache redirection in an edge deployment topology, the NetScaler sends all cacheable HTTP traffic to a transparent cache farm. Clients access the Internet through the NetScaler, which is configured as a Layer 4 switch that receives traffic on port 80.

The NetScaler can direct requests for images (for example, .gif and .jpg files) to one server in the transparent cache farm, and all other requests for static content to other servers in the farm. For this configuration, you configure content switching policies to send images to the image cache and send all other cacheable content to a default cache.

Note: The configuration described here is for transparent selective cache redirection. Therefore, it does not require a load balancing virtual server for the origin, as would a reverse proxy configuration.
To configure this type of selective cache redirection, first enable cache redirection, load balancing, and content switching. Then, configure a load balancing virtual server for the cache and configure an associated HTTP service. Then, configure a cache redirection virtual server and create and bind both cache redirection and content switching policies to this virtual server.

For details on how to enable cache redirection and load balancing on the NetScaler, see "Configuring Cache Redirection."

Enabling Content Switching

Nov 08, 2013

To configure selective cache redirection, after you enable both the load balancing and cache redirection features on the NetScaler, you must enable content switching.

To enable content switching by using the command line interface

At the command prompt, type:

- enable ns feature CS
- show ns feature

Example

```
> enable ns feature cs
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection         SP              ON
 3)    Load Balancing           LB              ON
 4)    Content Switching        CS              ON
 5)    Cache Redirection        CR              ON
       ...
       ...
       ...
 23)   HTML Injection           HTMLInjection   ON
 24)   NetScaler Push           push            OFF
Done
```

To enable cache redirection and load balancing by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Configure basic features.
3. In Configure Basic Features dialog box, select the check box next to the Content Switching, and then click OK.
4. In Enable/Disable Feature(s)? dialog box, click Yes.

Configuring a Load Balancing Virtual Server for the Cache

Dec 17, 2013

Create a load balancing virtual server and an HTTP service for each type of cache server that will be used. For example, if you want to serve JPEG files from one cache server and GIF files from another cache server, and use a third cache server for the rest of the content, create an HTTP service and virtual server for each of the three types of cache servers. Then bind each service to its respective virtual server.

For details on how to create a load balancing virtual server, see "Creating a Virtual ServerCreating a Virtual Server."

For details on how to configure services that represent the cache server, see "Configuring an HTTP Service."

For details on how to bind the service to a virtual server, see "Binding/Unbinding a Service to/from a Load Balancing Virtual Server."

For details on how to create a transparent proxy cache redirection server, see "Configuring a Cache Redirection Virtual Server", and create a virtual server of type TRANSPARENT.

For details on binding built-in cache redirection policies to the cache redirection virtual server, see "Binding Policies to the Cache Redirection Virtual Server."

Configuring a Cache Redirection Policy for a Specific Type of Content

To identify requests that contain a .gif or .jpeg extension as cacheable, you configure a cache redirection policy and bind it to the cache redirection virtual server.

Note: If a request matches a policy, the NetScaler appliance forwards it to the origin server. As a result, in the following procedure, you configure policies to match requests that do *not* have ".gif" or ".jpeg" extensions.
To configure cache redirection for a specific type of content, configure a policy that uses a simple expression, as described in "Configuring a Cache Redirection Policy."

Configuring Policies for Content Switching

Dec 17, 2013

You must create a content switching policy to identify specific types of content to be cached in one cache server or farm and identify other types of content to serve from another cache server or farm. For example, you can configure a policy to determine the location for image files with .gif and .jpeg extensions.

After defining the content switching policy, you bind it to a cache redirection virtual server and specify a load balancing virtual server. Requests that match the policy are forwarded to the named load balancing virtual server. Requests that do not match the content switching policy are forwarded to the default load balancing virtual server for the cache.

For more details about the content switching feature and configuring content switching policies, see "Content Switching."

You must first create the content switching policy and then bind it to the cache redirection virtual server.

To create a content switching policy by using the command line interface

At the command line, type:

- add cs policy <policyName> [-url <string> | -rule <expression>]
- show cs policy [<policyName>]

Examples

```
> add cs policy Policy-CS-JPEG -rule "REQ.HTTP.URL == '/*.jpeg'"
 Done
> show cs policy Policy-CS-JPEG
       Rule: REQ.HTTP.URL == '/*.jpeg'        Policy: Policy-CS-JPEG
       Hits: 0
 Done
>

> add cs policy Policy-CS-GIF -rule "REQ.HTTP.URL == '/*.gif'"
 Done
> show cs policy Policy-CS-GIF
       Rule: REQ.HTTP.URL == '/*.gif'        Policy: Policy-CS-GIF
       Hits: 0
 Done
>

> add cs policy Policy-CS-JPEG-URL -url /*.jpg
 Done
> show cs policy Policy-CS-JPEG-URL
       URL: /*.jpg     Policy: Policy-CS-JPEG-URL
       Hits: 0
 Done
>
```

```
> add cs policy Policy-CS-GIF-URL -url /*.gif
 Done
> show cs policy Policy-CS-GIF-URL
      URL: /*.gif    Policy: Policy-CS-GIF-URL
      Hits: 0
 Done
>
```

To create a URL-based content switching policy by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, click Add.
3. In the Create Content Switching Policy dialog box, in the Name text box, type a name for the policy.
4. Select the URL radio button.
5. In the Value text box, type the string value (for example, **/sports**).
6. Click Create and click Close. The policy you created appears in the Content Switching Policies page.

To create a rule-based content switching policy by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, click Add.
3. In the Create Content Switching Policy dialog box, in the Name text box, type a name for the policy.
4. Select the Expression radio button, and then click Configure.
5. In the Create Expression dialog box, choose the expression syntax that you want to use.
   - If you want to use default syntax, accept the default and proceed to the next step.
   - If you want to use classic syntax, click Switch to Classic Syntax.

   The Expression portion of the dialog box changes to match your choice. The default syntax Expression view has fewer elements than does the classic syntax Expression view. In the default syntax Expression view, instead of a preview window, a button provides access to an expression evaluator. The evaluator evaluates the expression you entered, to verify that it is valid, and displays an analysis of the expression's effect.

6. Enter your policy expressions.
   - If you are using classic syntax and need further instructions, see "Configuring Classic Policies and Expressions."
   - If you are using the default syntax and need further instructions, see "Configuring Default Syntax Expressions: Getting Started."
7. Click Create and click Close. The policy you created appears in the Content Switching Policies pane.

To bind the content switching policy to a cache redirection virtual server by using the command line interface

At the command prompt, type the following commands to bind the content switching policy to a cache redirection virtual server and verify the configuration:

- bind cs vserver <name> <targetVserver> [-policyName <string>]
- show cs vserver [<name>]

Example

```
> bind cs vserver Vserver-CR-1 lbcachejpeg -policyName Policy-CS-JPEG
 Done
```

```
> bind cs vserver Vserver-CR-1 lbcachegif -policyName Policy-CS-GIF
 Done
> show cs vserver Vserver-CR-1
      Vserver-CR-1 (10.102.29.60:80) - HTTP   Type: CONTENT
      State: UP
      Last state change was at Fri Jul  2 12:53:45 2010
      Time since last state change: 0 days, 00:00:58.920
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Port Rewrite : DISABLED
      State Update: DISABLED
      Default:       Content Precedence: RULE
      Cacheable: YES
      Vserver IP and Port insertion: OFF
      Case Sensitivity: ON
      Push: DISABLED  Push VServer:
      Push Label Rule: none

1)     Policy: Policy-CS-JPEG  Target: lbcachejpeg     Priority: 0     Hits: 0
2)     Policy: Policy-CS-GIF   Target: lbcachegif      Priority: 0     Hits: 0
 Done
>
```

To bind the content switching policy to a cache redirection virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the policy (for example, **Vserver-CS-1**), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Policies tab, click CSW, and then click Insert Policy.
4. In the Policy Name column, select the policy that you want to configure for the content switching virtual server.
5. In the Target column, click the green arrow, and select the target load balancing virtual server from the list.
6. Click OK.

Configuring Precedence for Policy Evaluation

Aug 23, 2013

You can configure a content switching policy based on either a rule, which is a generic configuration to accommodate various content types, or a URL, which is more specific and defines exactly the type of content that has to be sent to a particular cache server. Essentially, the same content can be defined by either a rule based policy or a URL based policy.

Once you bind content switching policies of either type to a cache redirection virtual server, you can configure the virtual server to give precedence to either rule based or URL based policies. This will, in turn, decide which servers the particular requests are directed to.

To configure precedence for policy evaluation, use the precedence parameter, which specifies the type of policy (URL or RULE) that takes precedence on the content redirection virtual server.

Possible values: RULE, URL

Default value: RULE

To configure precedence for policy evaluation by using the command line interface

At the command prompt, type the following commands to configure precedence for policy evaluation and verify the configuration:

- set cr vserver <name> [-precedence (RULE | URL)]
- show cr vserver <name>

Example

```
> set cr vserver Vserver-CRD-1 -precedence URL
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: Policy-CRD
 Done
>
```

To configure precedence for policy evaluation by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure precedence, (for example, **Vserver-CS-1**), and then click Open.

3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, next to Precedence, click Rule or URL, and then click OK.

Administering a Cache Redirection Virtual Server

Mar 30, 2012

To administer a cache redirection virtual server, you need to view cache redirection statistics. You might need to enable or disable cache redirection servers, or direct policy hits to the cache instead of the origin. Administrative tasks also include backing up a cache redirection virtual server and managing client connections.

Viewing Cache Redirection Virtual Server Statistics

Aug 23, 2013

You can view properties of a cache redirection virtual server and statistics on the traffic that has passed through a cache redirection virtual server. You can also view the cache redirection virtual servers and policies that you have bound to load balancing virtual servers.

To view statistics for a specific cache redirection virtual servers, use the name parameter to specify the name of the virtual server for which statistics will be displayed. Otherwise, statistics for all cache redirection virtual servers are displayed.
Maximum Length: 127

To view statistics for a cache redirection virtual server by using the command line interface

At the command prompt, type:

stat cr vserver [<name>]

Example

```
> stat cr vserver Vserver-CRD-1

Vserver Summary
                 IP  port    Protocol      State
Vser...CRD-1      0.0.0.0   80       HTTP         UP

VServer Stats:
                           Rate (/s)          Total
Requests                          0                0
Responses                         0                0
Request bytes                     0                0
Response bytes                    0                0

 Done
>
```

To view statistics for a cache redirection virtual server by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers
2. In the details pane, select the virtual server for which you want to view statistics, (for example, **Vserver-CRD-1**), and then click Statistics.

Omit the server name to display basic statistics for all cache redirection virtual servers. Include the server name to display detailed statistics for that virtual server, including number and size of requests and responses that pass through the virtual server

To view the statistics of a cache redirection virtual server by using the monitoring and dashboard utilities

1. To view the statistics by using the monitoring utilities, click the Monitoring tab.
2. In the Select Group drop-down menu, choose CR Virtual Servers. A list of cache redirection virtual servers appears.
3. To view the statistics by using the dashboard utilities, click the Dashboard tab.

4. Click Applet Client or Web Start Client next to Statistical Utility.
5. In the Select Group drop-down menu, choose CR Virtual Servers. The dashboard displays summary statistics for the cache redirection virtual servers.
6. To see a chart of virtual server activity, click Chart. A graphical representation of the virtual server statistics appears.

Enabling or Disabling a Cache Redirection Virtual Server

Aug 23, 2013

When you create a cache redirection virtual server, it is enabled by default. If you disable a cache redirection virtual server, its state changes to OUT OF SERVICE and it stops redirecting cacheable client requests. However, the NetScaler appliance continues to respond to ARP and ping requests for the IP address of this virtual server.

To Enable or Disable a cache redirection virtual servers by using the command line interface

At the command line, type one of the following commands:

- enable cr vserver <name>
- show cr vserver <name>
- disable cr vserver <name>
- show cr vserver <name>

Examples

```
> enable cr vserver Vserver-CRD-1
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: Policy-CRD
 Done
>
```

```
> disable cr vserver Vserver-CRD-1
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: OUT OF SERVICE  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: ORIGIN L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF
```

1) Cache bypass Policy: bypass-cache-control
2) Cache bypass Policy: Policy-CRD
Done
>

To Enable or Disable a cache redirection virtual servers by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the navigation pane, expand Cache Redirection, and then click Virtual Servers.
3. In the details pane, select the virtual server that you want to enable or disable, (for example, **Vserver-CRD-1**), and then click Statistics.
4. In the Proceed dialog box, click Yes.

Directing Policy Hits to the Cache Instead of the Origin

Aug 23, 2013

By default, when a request matches a policy, the NetScaler appliance forwards the request either to the origin server directly, or to a load balancing virtual server for the origin, depending on how you have configured cache redirection.

You can change the default behavior so that when a request matches a policy, the request is forwarded to a load balancing virtual server for the cache.

To change the destination for a policy hit to the origin or the cache, use the onPolicyMatch parameter, which specifies where to send requests that match the cache redirection policy.

The valid options are:

1. CACHE - Directs all matching requests to the cache.
2. ORIGIN - Directs all matching requests to the origin server.

Note: For this option to work, you must select the cacheredirection type as POLICY.
Possible values: CACHE, ORIGIN

Default value: ORIGIN

To change the destination for a policy hit to the origin or the cache by using the command line interface

At the command prompt, type the following commands to change the destination for a policy hit and verify the configuration:

- set cr vserver <name> [-onPolicyMatch (ORIGIN | CACHE)]
- show cr vserver <name>

Example

```
> set cr vserver Vserver-CRD-1 -onPolicyMatch CACHE
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: CACHE  L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: Policy-CRD
 Done
```

To change the destination for a policy hit to the origin or the cache by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, select the virtual server for which you want to change the destination for a policy hit, (for example, **Vserver-CRD-1**), and then click Open.
3. In Configure Virtual Server (Cache Redirection) dialog box, click Advanced tab.
4. Select CACHE or ORIGIN from the Redirect To drop-down list.
5. Click OK.

Backing Up a Cache Redirection Virtual Server

Aug 22, 2013

Cache redirection can fail if the primary virtual server fails, or if it is unable to handle excessive traffic. You can specify a backup virtual server to take over the processing of traffic when the primary virtual server fails.

To specify a backup cache redirection virtual server, use the backupVServer parameter, which specifies Backup Virtual Server. Maximum Length: 127

To specify a backup cache redirection virtual server by using the command line interface

At the command prompt, type the following commands to specify a backup cache redirection virtual server and verify the configuration:

- set cr vserver <name> [-backupVServer <string>]
- show cr vserver <name>

Example

```
> set cr vserver Vserver-CRD-1 -backupVServer Vserver-CRD-2
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: CACHE  L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF
      Backup: Vserver-CRD-2

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: Policy-CRD
 Done
```

To specify a backup cache redirection virtual server by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > irtual Servers.
2. In the details pane, select the virtual server for which you want to change the destination for a policy hit, (for example, **Vserver-CRD-1**), and then click Open.
3. In Configure Virtual Server (Cache Redirection) dialog box, select the Advanced tab.
4. In the Backup Virtual Server drop-down list, select the virtual server.
5. Click OK.

Managing Client Connections for a Virtual Server

Jun 04, 2015

You can configure timeouts on a cache redirection virtual server so that client connections are not kept open indefinitely. You can also insert Via headers in requests. To possibly reduce network congestion, you can reuse open TCP connections. You can enable or disable delayed cleanup of cache redirection virtual server connections.

You can configure the NetScaler to send ICMP responses to PING requests according to your settings. On the IP address corresponding to the virtual server, set the ICMP RESPONSE to VSVR_CNTRLD, and on the virtual server, set the ICMP VSERVER RESPONSE.

The following settings can be made on a virtual server:

- When you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.
- When you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds even if one virtual server is UP.
- When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds even if one virtual server set to ACTIVE is UP.

This document includes the following information:

- Configuring Client Timeout
- Inserting Via Headers in the Requests
- Reusing TCP Connections
- Configuring Delayed Connection Cleanup

Updated: 2013-08-22

You can specify expiration of client requests by setting a timeout value for the cache redirection virtual server. The timeout value is the number of seconds for which the cache redirection virtual server waits to receive a response for the client request.

To configure a time-out value, use the cltTimeout parameter, which specifies the time, in seconds, after which the NetScaler appliance closes any idle client connections. The default value is 180sec for HTTP/SSL-based services and 9000sec for TCP-based services.

To configure client timeout by using the command line interface

At the command prompt, type the following commands to configure client timeout and verify the configuration:

- set cr vserver <name> [-cltTimeout <secs>]
- show cr vserver <name>

Example

```
> set cr vserver Vserver-CRD-1 -cltTimeout 6000
 Done
```

```
> show cr vserver Vserver-CRD-1
        Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
        State: UP  ARP:DISABLED
        Client Idle Timeout: 6000 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Default:        Content Precedence: URL Cache: TRANSPARENT
        On Policy Match: CACHE  L2Conn: OFF     OriginUSIP: OFF
        Redirect: POLICY        Reuse: ON       Via: ON ARP: OFF
        Backup: Vserver-CRD-2

1)      Cache bypass  Policy: bypass-cache-control
2)      Cache bypass  Policy: Policy-CRD
 Done
```

To configure client timeout by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure client timeout, (for example, **Vserver-CRD-1**), and then click Open.
3. In Configure Virtual Server (Cache Redirection) dialog box, select the Advanced tab.
4. In the Client Time-out(secs) text box, enter the time-out value in seconds.
5. Click OK.

Updated: 2013-08-23

A Via header lists the protocols and recipients between the start and end points for a request or a response and informs the server of proxies through which the request was sent. You can configure the cache redirection virtual server to insert a Via header in each HTTP request. The via parameter is enabled by default when you create a cache redirection virtual server.

To enable or disable Via-header insertion in client requests, use the via parameter, which specifies the state of the system in inserting a Via header in the HTTP requests.

Possible values: ON, OFF

Default value: ON

To enable or disable Via-header insertion in client requests by using the command line interface

At the command prompt, type:

- set cr vserver <name> [-via (ON | OFF)]
- show cr vserver <name>

Example

```
> set cr vserver Vserver-CRD-1 -via ON
```



```
 Done
> show cr vserver Vserver-CRD-1
        Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
        State: UP  ARP:DISABLED
        Client Idle Timeout: 6000 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Default:        Content Precedence: URL Cache: TRANSPARENT
        On Policy Match: CACHE  L2Conn: OFF     OriginUSIP: OFF
        Redirect: POLICY        Reuse: ON       Via: ON ARP: OFF
        Backup: Vserver-CRD-2

1)      Cache bypass  Policy: bypass-cache-control
2)      Cache bypass  Policy: Policy-CRD
 Done
>
```

To enable or disable Via-header insertion in client requests by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure client timeout, (for example, **Vserver-CRD-1**), and then click Open.
3. In Configure Virtual Server (Cache Redirection) dialog box, select the Advanced tab.
4. Select the Via check box.
5. Click OK.

Updated: 2013-11-08

You can configure the NetScaler appliance to reuse TCP connections to the cache and origin servers across client connections. This can improve performance by saving the time required to establish a session between the server and the NetScaler. The reuse option is enabled by default when you create a cache redirection virtual server.

To enable or disable the reuse of TCP connections, use the reuse parameter, which specifies the state of reuse of TCP connections to the cache or origin servers across client connections.

Possible values: ON, OFF

Default value: ON

To enable or disable the reuse of TCP connections by using the command line interface

At the command prompt, type:

- set cr vserver <name> [-reuse (ON|OFF)]
- show cr vserver <name>

Example


```
> set cr vserver Vserver-CRD-1 -reuse ON
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 6000 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: CACHE  L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF
      Backup: Vserver-CRD-2

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: Policy-CRD
 Done
```

To enable or disable the reuse of TCP connections by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure client timeout, (for example, **Vserver-CRD-1**), and then click Open.
3. In Configure Virtual Server (Cache Redirection) dialog box, select the Advanced tab.
4. Select the Reuse check box.
5. Click OK.

Updated: 2013-08-22

The down state flush option performs delayed cleanup of connections on a cache redirection virtual server. The down state flush option is enabled by default when you create a cache redirection virtual server.

To enable or disable the down state flush option, set the downStateFlush parameter.

Possible values: ENABLED, DISABLED

Default value: ENABLED

To enable of disable the down state flush option by using the command line interface

At the command prompt, type the following commands to configure delayed connection clean up and verify the configuration:

- set cr vserver <name> [-downStateFlush (ENABLED | DISABLED)]
- show cr vserver <name>

Example

```
> set cr vserver Vserver-CRD-1 -downStateFlush ENABLED
 Done
> show cr vserver Vserver-CRD-1
      Vserver-CRD-1 (*:80) - HTTP     Type: CONTENT
      State: UP  ARP:DISABLED
      Client Idle Timeout: 6000 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      Default:       Content Precedence: URL Cache: TRANSPARENT
      On Policy Match: CACHE  L2Conn: OFF     OriginUSIP: OFF
      Redirect: POLICY       Reuse: ON       Via: ON ARP: OFF
      Backup: Vserver-CRD-2

1)     Cache bypass  Policy: bypass-cache-control
2)     Cache bypass  Policy: Policy-CRD
 Done
```

To enable or disable the reuse of TCP connections by using the configuration utility

1. Navigate to Traffic Management > Cache Redirection > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure client timeout, (for example, **Vserver-CRD-1**), and then click Open.
3. In Configure Virtual Server (Cache Redirection) dialog box, click Advanced tab.
4. Select the Down state flush check box.
5. Click OK.

N-Tier Cache Redirection

Mar 30, 2012

To efficiently handle large amounts of cached data, typically several gigabytes per second, an Internet Service Provider (ISP) deploys several dedicated cache servers. The cache redirection feature of the NetScaler appliance can help load balance the cache servers, but a single appliance or a couple of appliances might not efficiently handle the large volume of traffic.

You can solve the problem by deploying the NetScaler appliances in two tiers (layers), where the appliances in the upper tier load balance those in the lower tier and the appliances in the lower tier load balance the cache servers. This arrangement is called *n-tier cache redirection*.

For purposes such as auditing and security, an ISP has to track client details such as the IP address, information provided, and the time of the interaction. Therefore, client connections through a NetScaler appliance have to be fully transparent. However, if you configure transparent cache redirection, with the NetScaler appliances deployed in parallel, the IP address of the client has to be shared among all the appliances. Sharing of the client IP address creates a conflict that makes network devices, such as routers, cache servers, origin servers, and other NetScaler appliances, unable to determine the appliance, and therefore the client, to which the response should be sent.

To solve the problem, NetScaler n-tier cache redirection splits the source port range among the appliances in the lower tier and includes the client IP address in the request sent to the cache servers. The upper-tier NetScaler appliances are configured to do sessionless load balancing in order to avoid unnecessary load on the appliances.

When the lower-tier NetScaler appliance communicates with a cache server, it uses a mapped IP address (MIP) to represent the source IP address. Therefore, the cache server can identify the NetScaler from which it received the request and send the response to the same NetScaler.

The lower-tier NetScaler appliance inserts the client IP address into the header of the request sent to the cache server. The client IP in the header helps the NetScaler to determine the client to which the packet should be forwarded when it receives the response from a cache server, or the origin server in case of a cache miss. The origin server determines the response to be sent according to the client IP inserted in the request header.

The origin server sends the response to an upper-tier NetScaler, including the source port number from which the origin server received the request. The entire source port range, 1024 to 65535, is distributed among the lower-tier NetScaler appliances. Each lower-tier appliance is exclusively assigned a group of addresses within the range. This allotment enables the upper-tier appliance to unambiguously identify the lower-tier NetScaler appliance that sent the request to the origin server. The upper-tier appliance can therefore forward the response to the correct lower-tier appliance.

The upper-tier NetScaler appliances are configured to do policy-based routing, and the routing policies are defined to determine the IP address of the destination NetScaler from the source port range.

The following setup is necessary for the functioning of n-tier cache redirection:

For each upper-tier NetScaler appliance:

- Enable Layer 3 mode.
- Define policies for policy-based routes (PBRs) so that traffic is forwarded according to the range of the destination

port.

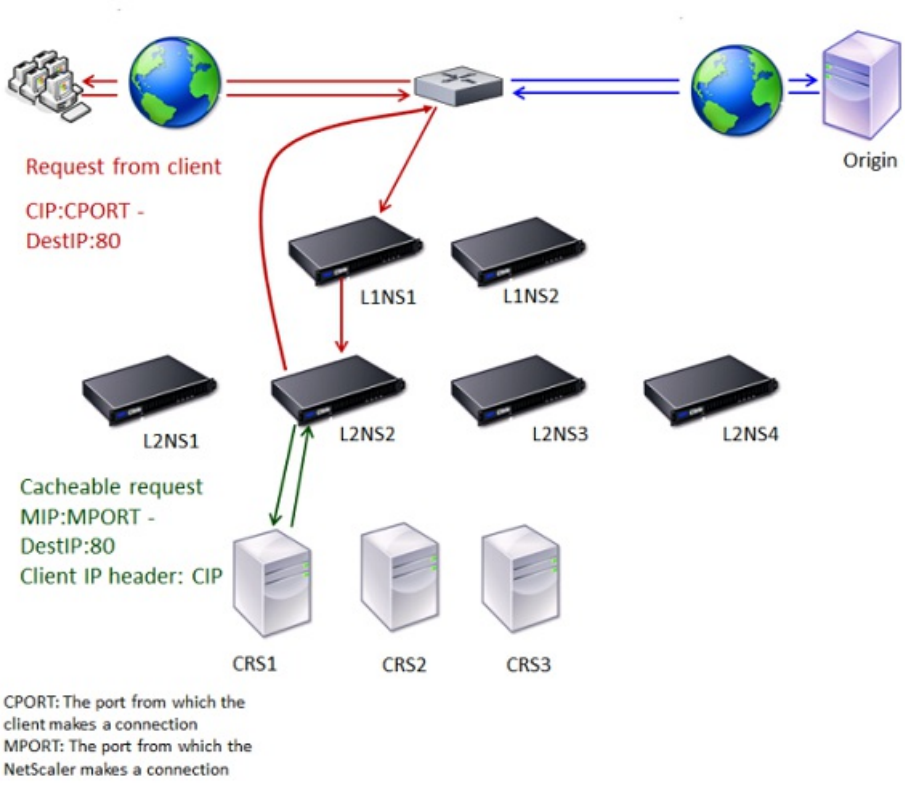
- Configure a load balancing virtual server.
- Configure the virtual server to listen to all the traffic coming from the client. Set the Service Type/Protocol to be ANY and IP Address as asterisk (*).
- Enable sessionless load balancing with MAC-based redirection mode to avoid unnecessary load on the upper-tier NetScaler appliances.
- Make sure that the Use Proxy Port option is enabled.
- Create a service for each lower-tier NetScaler and bind all the services to the virtual server.

For each lower-tier NetScaler appliance,

- Configure the cache redirection port range on the NetScaler. Assign an exclusive range to each lower-tier NetScaler.
- Configure a load balancing virtual server and enable MAC-based redirection.
- Create a service for each cache server that is to be load balanced by this NetScaler. When creating the service, enable insertion of client IP in the header. Then, bind all the services to the load balancing virtual server.
- Configure a transparent mode cache redirection virtual server with the following settings:
  - Enable the Origin USIP option.
  - Add a source IP expression to include the client IP in the header.
  - Enable the Use Port Range option.

The following figure shows how cache redirection works when a client request is cacheable and the response is sent from a cache server.

Figure 1. Cache Redirection in Case of a Cache Hit



Two NetScaler appliances, L1NS1 and L1NS2, are deployed in the upper tier, and four NetScaler appliances, L2NS1, L2NS2,

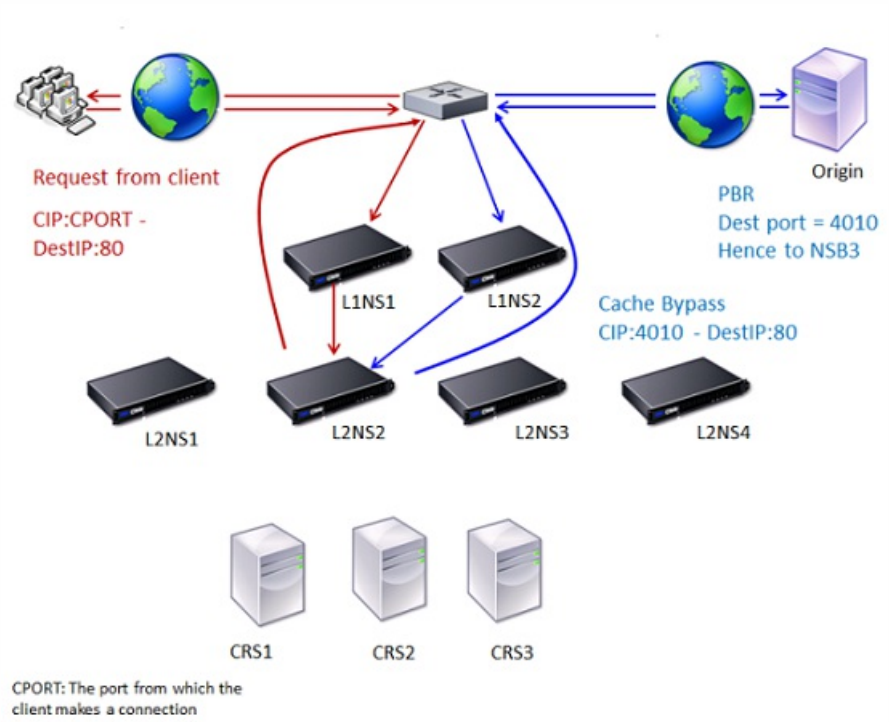
L2NS3, and L2NS4, are deployed in the lower tier. Client A sends a request, which is forwarded by the router. Cache servers CRS1, CRS2, and CRS3 service the cache requests. Origin Server O services the uncached requests.

Traffic Flow

1. Client sends a request, and the router forwards it to L1NS1.
2. L1NS1 load balances the request to L2NS2.
3. L2NS2 load balances the request to the cache server CRS1, and the request is cacheable. L2NS2 includes the client IP in the request header.
4. CRS1 sends the response to L2NS2 because L2NS2 used its MIP as the source IP address when connecting to CRS1.
5. With the help of the client IP address in the request header, L2NS2 identifies the client from which the request came. L2NS2 directly sends the response to the router, avoiding unnecessary load on the NetScaler in the upper tier.
6. The router forwards the response to Client A.

The following figure shows how cache redirection works when a client request is sent to an origin server for a response.

Figure 2. Cache Redirection in Case of a Cache Bypass



Two NetScaler appliances, L1NS1 and L1NS2, are deployed in the upper tier, and four NetScaler appliances, L2NS1, L2NS2, L2NS3, and L2NS4, are deployed in the lower tier. Client A sends a request, which is forwarded by the router. Cache servers CRS1, CRS2, and CRS3 service the cache requests. Origin Server O services the uncached requests.

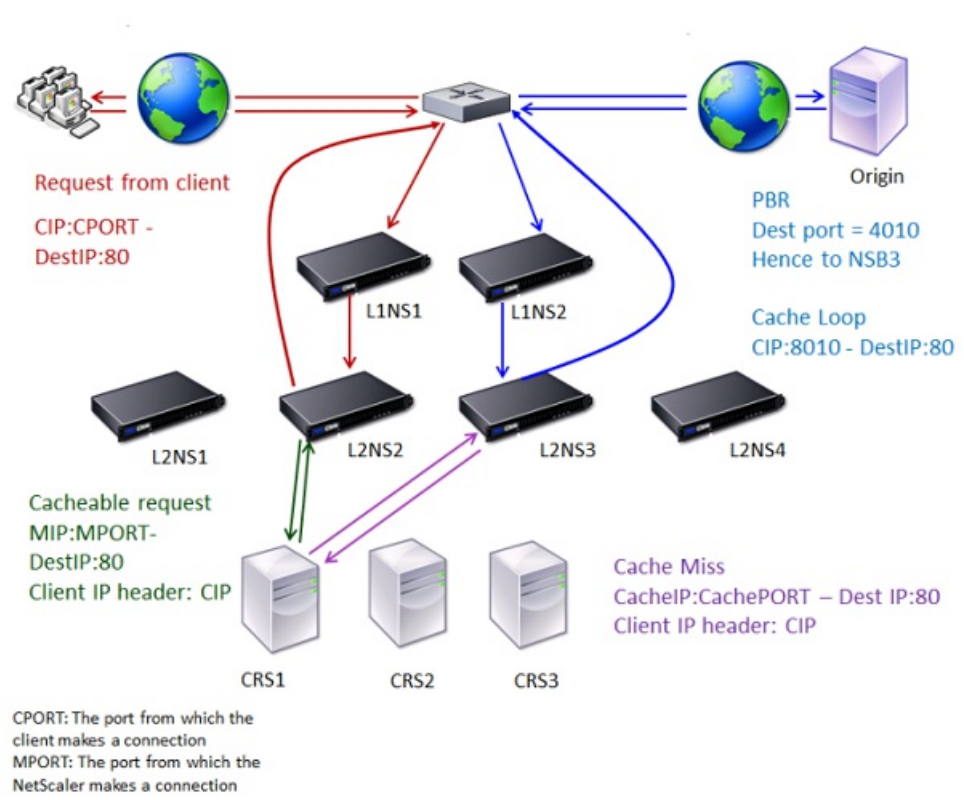
Traffic Flow

1. Client sends a request, and the router forwards it to L1NS1.
2. L1NS1 load balances the request to L2NS2.
3. The request is uncacheable (cache bypass). Therefore, L2NS2 sends the request to the origin server through the router.
4. The origin server sends the response to an upper-tier NetScaler, L1NS2.

5. According to the PBR policies, L1NS2 forwards the traffic to the appropriate NetScaler in the lower tier, L2NS2.
6. L2NS2 uses the client IP address in the request header to identify the client from which the request came and sends the response directly to the router, avoiding unnecessary load on the NetScaler in the upper tier.
7. The router forwards the response to Client A.

The following figure shows how cache redirection works when a client request is not cached.

Figure 3. Cache Redirection in Case of a Cache Miss



Two NetScaler appliances, L1NS1 and L1NS2, are deployed in the upper tier, and four NetScaler appliances, L2NS1, L2NS2, L2NS3, and L2NS4, are deployed in the lower tier. Client A sends a request, which is forwarded by the router. Cache servers CRS1, CRS2, and CRS3 service the cache requests. Origin Server O services the uncached requests.

Traffic Flow

1. Client sends a request, and the router forwards it to L1NS1.
2. L1NS1 load balances the request to L2NS2.
3. L2NS2 load balances the request to the cache server CRS1 because the request is cacheable.
4. CRS1 does not have the response (cache miss). CRS1 forwards the request to the origin server through the NetScaler in the lower tier. L2NS3 intercepts the traffic.
5. L2NS3 takes the client IP from the header and forwards the request to the origin server. The source port included in the packet is the L2NS3 port from which the request is sent to the origin server.
6. The origin server sends the response to an upper-tier NetScaler, L1NS2.
7. According to the PBR policies, L1NS2 forwards the traffic to the appropriate NetScaler in the lower tier, L2NS3.
8. L2NS3 forwards the response to the router.
9. The router forwards the response to Client A.

Configuring the Upper-Tier NetScaler Appliances

Oct 31, 2013

Configure each of the upper-tier NetScaler appliances as follows.

At the command prompt, type the following commands:

- add service <name>@ <serviceIP> <serviceType> <port>
  Run this command for each service to be added.
- add lb vserver <name>@ ANY * <port> -persistenceType <persistenceMethod> -lbMethod <lbMethod> -m MAC -sessionless ENABLED -cltTimeout <client_Timeout_Value>
- bind lb vserver <name>@ <serviceName>
  Run this command for each service to be bound.
- enable ns mode l3
- add ns pbr <name> <action> -srcPort <sourcePortNumber> -destPort <startPortNumber-endPortNumber> -nexthop <serviceIpAddress> -protocol TCP
- apply ns pbrs

  Run this command after adding all the necessary PBRs.

1. Enable L3 mode:
   1. In the navigation pane, click System, and then click Settings.
   2. In the Settings group, click the Configure modes link.
   3. Select the Layer 3 Mode (IP Forwarding) check box.
   4. Click OK.
2. Configure policy-based routing (PBR):
   1. Navigate to System > Network > PBRs.
   1. In the Policy-Based Routing (PBRs) pane, click Add.
   2. Type a name for the PBR.
   3. Select the action as Allow.
   4. In the Next Hop box, type the IP address of the service, which represents a lower-tier NetScaler.
   5. Select TCP from the Protocol drop-down list.
   6. Type the source port and the range of the destination port corresponding to the lower-tier NetScaler being added.
   7. Click Create.
   8. In the details pane, select the PBR and click Apply.
   9. Repeat Step (i) to Step (vii) for each lower-tier NetScaler.
3. Create a service for each lower-tier NetScaler:
   1. Navigate to Traffic Management > Load Balancing > Services.
   1. In the details pane, click Add.
   2. Specify the name, protocol, IP address, and port. The protocol should be ANY.
   3. Click Create.

4. Configure a load balancing virtual server:
    1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
    1. In the details pane, click Add.
    2. Specify the name, protocol, IP address, and port. The protocol should be ANY and the IP address should be *.
    3. In the Services tab, select the services that represent the lower-tier NetScaler appliances.
    4. In the Advanced tab, select the Redirection Mode as MAC Based and select the Sessionless check box.
    5. Click Create.

Configuring the Lower-Tier NetScaler Appliances

Oct 31, 2013

Configure each of the lower-tier NetScaler appliances as follows.

At the command prompt, type the following commands:

- add service <name>@ <cacheServiceIP> <serviceType> <port> -cip ENABLED "ClientIP" – cachetype transparent
  Repeat for each cache server.
- add lb vserver <name>@ <serviceType> -m MAC
- bind lb vserver <name>@ <cacheServiceName>
  Repeat for each cache server.
- add cr vserver <name> <serviceType> * <port> -srcIPExpr "HTTP.REQ.HEADER(\"ClientIP\")" -originusip ON – usePortRange ON
- set ns param-crPortRange <startPortNumber-endPortNumber>

1. Create a service for each cache server. To create a service:
   1. Navigate to Traffic Management > Load Balancing > Services.
   1. In the details pane, click Add, and specify the name and protocol. Clear the Directly Addressable check box.
   2. In the Advanced tab, select the Override Global check box and the Client IP check box, and then in the Header box, type ClientIP.
   3. In the Cache Type box, select Transparent Cache.
   4. Click Create.
2. Configure a load balancing virtual server:
   1. Navigate to Traffic Management > Load Balancing > Virtual Services.
   1. In the details pane, click Add and specify the name, protocol, IP address, and port. The IP address should be an asterisk (*).
   2. In the Services tab, select the services that represent the cache servers.
   3. In the Advanced tab, for Redirection Mode, select MAC Based.
   4. Click Create.
3. Configure a cache redirection virtual server:
   1. Navigate to Traffic Management > Load Balancing > Virtual Services.
   1. In the details pane, click Add and specify the name, protocol, IP address, and port. The IP address should be *.
   2. For Cache Type, select Transparent.
   3. On the Advanced tab, in the Cache Server box, select the new load balancing virtual server and check the Origin USIP and Use Port Range check boxes. In the Source IP Expression box, type HTTP.REQ.HEADER("ClientIP").
   4. Click Create.
4. Assign a source port range for the NetScaler:
   1. In the navigation pane, click System, and then click Settings.
   2. In the Settings group, click the Change global system settings link.

3. In the Cache Redirection Port Range group, specify the port range for the NetScaler by typing a port number for Start Port and a port number for End Port.
4. Click OK.

Content Switching

May 25, 2015

In today's complex Web sites, you may want to present different content to different users. For example, you may want to allow users from the IP range of a customer or partner to have access to a special Web portal. You may want to present content relevant to a specific geographical area to users from that area. You may want to present content in different languages to the speakers of those languages. You may want to present content tailored to specific devices, such as smartphones, to those who use the devices. The Citrix NetScaler content switching feature enables the NetScaler appliance to distribute client requests across multiple servers on the basis of specific content that you wish to present to those users.

To configure content switching, first create a basic content switching setup, and then customize it to meet your needs. This entails enabling the content switching feature, setting up load balancing for the server or servers that host each version of the content that is being switched, creating a content switching virtual server, creating policies to choose which requests are directed to which load balancing virtual server, and binding the policies to the content switching virtual server. You can then customize the setup to meet your needs by setting precedence for your policies, protecting your setup by configuring a backup virtual server, and improving the performance of your setup by redirecting requests to a cache.

Updated: 2013-08-22

Content Switching enables the NetScaler appliance to direct requests sent to the same Web host to different servers with different content. For example, you can configure the appliance to direct requests for dynamic content (such as URLs with a suffix of .asp, .dll, or .exe) to one server and requests for static content to another server. You can configure the appliance to perform content switching based on TCP/IP headers and payload.

You can also use content switching to configure the appliance to redirect requests to different servers with different content on the basis of various client attributes. Some of those client attributes are:

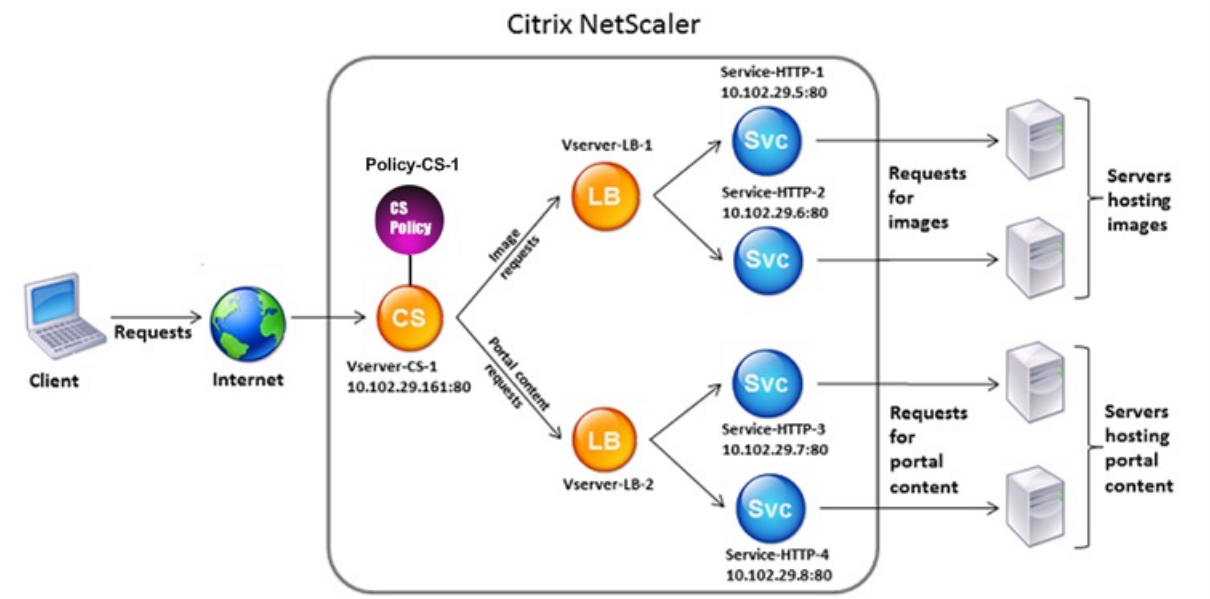
- **Device Type.** The appliance examines the user agent or custom HTTP header in the client request for the type of device from which the request originated. Based on the device type, it directs the request to a specific Web server. For example, if the request came from a cell phone, the request is directed to a server that is capable of serving content that the user can view on his or her cell phone. A request from a computer is directed to a different server that is capable of serving content designed for a computer screen.
- **Language.** The appliance examines the Accept-Language HTTP header in the client request and determines the language used by the client's browser. The appliance then sends the request to a server that serves content in that language. For example, using content switching based on language, the appliance can send someone whose browser is configured to request content in French to a server with the French version of a newspaper. It can send someone else whose browser is configured to request content in English to a server with the English version.
- **Cookie.** The appliance examines the HTTP request headers for a cookie that the server set previously. If it finds the cookie, it directs requests to the appropriate server, which hosts custom content. For example, if a cookie is found that indicates that the client is a member of a customer loyalty program, the request is directed to a faster server or one with special content. If it does not find a cookie, or if the cookie indicates that the user is not a member, the request is directed to a server for the general public.
- **HTTP Method.** The appliance examines the HTTP header for the method used, and sends the client request to the right server. For example, GET requests for images can be directed to an image server, while POST requests can be directed to

a faster server that handles dynamic content.

- **Layer 3/4 Data.** The appliance examines requests for the source or destination IP, source or destination port, or any other information present in the TCP or UDP headers, and directs the client request to the right server. For example, requests from source IPs that belong to customers can be directed to a custom web portal on a faster server, or one with special content.

A typical content switching deployment consists of the entities described in the following diagram.

Figure 1. Content Switching Architecture



A content switching configuration consists of a content switching virtual server, a load balancing setup consisting of load balancing virtual servers and services, and content switching policies. To configure content switching, you must configure a content switching virtual server and associate it with policies and load balancing virtual servers. This process creates a *content group*—a group of all virtual servers and policies involved in a particular content switching configuration.

Content switching can be used with HTTP, HTTPS, TCP, and UDP connections. For HTTPS, you must enable SSL Offload.

When a request reaches the content switching virtual server, the virtual server applies the associated content switching policies to that request. The priority of the policy defines the order in which the policies bound to the content switching virtual server are evaluated. If you are using default syntax policies, when you bind a policy to the content switching virtual server, you must assign a priority to that policy. If you are using NetScaler classic policies, you can assign a priority to your policies, but are not required to do so. If you assign priorities, the policies are evaluated in the order that you set. If you do not, the NetScaler appliance evaluates your policies in the order in which they were created.

In addition to configuring policy priorities, you can manipulate the order of policy evaluation by using Goto expressions and policy bank invocations. For more details about default syntax policy configuration, see "Configuring Default Syntax Policies."

After it evaluates the policies, the content switching virtual server routes the request to the appropriate load balancing virtual server, which sends it to the appropriate service.

Content switching virtual servers can only send requests to other virtual servers. If you are using an external load balancer,

you must create a load balancing virtual server for it and bind its virtual server as a service to the content switching virtual server.

Configuring Basic Content Switching

Jun 09, 2015

Before you configure content switching, you must understand how content switching is set up and how the services and virtual servers are connected.

To configure a basic, functional content switching setup, first enable the content switching feature. Then, create at least one content group. For each content group, create a content switching virtual server to accept requests to a group of web sites that use content switching. Also create a load balancing setup, which includes a group of load balancing virtual servers to which the content switching virtual server directs requests. To specify which requests to direct to which load balancing virtual server, create at least two content switching policies, one for each type of request that is to be redirected. When you have created the virtual servers and policies, bind the policies to the content switching virtual server. You can also bind a policy to multiple content switching virtual servers. When you bind a policy, you specify the load balancing virtual server to which requests that match the policy are to be directed.

In addition to binding individual policies to a content switching virtual server, you can bind policy labels. If you create additional content groups, you can bind a policy or policy label to more than one of the content switching virtual servers.

Note: After creating a content group, you can modify its content switching virtual serve to customize the configuration. For information on modifying the configuration of an existing content switching virtual server, see "Customizing the Basic Content Switching Configuration." For information on disabling and re-enabling entities, unbinding policies, and removing entities, see "Managing a Content Switching Setup."

This section includes the following details:

- Enabling Content Switching
- Creating Content Switching Virtual Servers
- Configuring a Load Balancing Setup for Content Switching
- Configuring a Content Switching Action
- Configuring Content Switching Policies
- Configuring Content Switching Policy Labels
- Binding Policies to a Content Switching Virtual Server
- Configuring Policy Based Logging for Content Switching
- Verifying the Configuration

Enabling Content Switching

Oct 31, 2013

To use the content switching feature, you must enable content switching. You can configure content switching entities even though the content switching feature is disabled. However, the entities will not work.

At the command prompt, type the following commands to enable content switching and verify the configuration:

- enable ns feature CS
- show ns feature

Example

```
> enable  feature ContentSwitch
 Done
> show feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection          SP              ON
 3)    Load Balancing           LB              ON
 4)    Content Switching        CS              ON
 .
 .
 .
 22)   Responder              RESPONDER         ON
 23)   HTML Injection           HTMLInjection     ON
 24)   NetScaler Push           push            OFF
Done
```

1. Navigate to System > Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Content Switching check box, and then click OK.
4. In the Enable/Disable Features(?) dialog box, click Yes.

Creating Content Switching Virtual Servers

Apr 07, 2014

You can add, modify, and remove content switching virtual servers. The state of a virtual server is DOWN when you create it, because the load balancing virtual server is not yet bound to it.

At the command prompt, type:

add cs vserver <name> <protocol> <IPAddress> <port>

Example

```
add cs vserver Vserver-CS-1 HTTP 10.102.29.161 80
```

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Content Switching) dialog box, in the Name, IP Address, and Port text boxes, type the name, IP address, and port of the virtual server, (for example, Vserver-CS-1, 10.102.29.161, and 80).
   Note: If you need to enter an IPv6 address, select the IPv6 check box before you enter the address.
4. In the Protocol list, select the type of the virtual server (for example, HTTP).
5. Click Create, and then click Close.

Configuring a Load Balancing Setup for Content Switching

Dec 17, 2013

The content switching virtual server redirects all requests to a load balancing virtual server. You must create one load balancing virtual server for each version of the content that is being switched. This is true even when your setup has only one server for each version of the content, and you are therefore not doing any load balancing with those servers. You can also configure actual load balancing with multiple load-balanced servers that mirror each version of the content. In either scenario, the content switching virtual server needs to have a specific load balancing virtual server assigned to each version of the content that is being switched.

The load balancing virtual server then forwards the request to a service. If it has only one service bound to it, it selects that service. If it has multiple services bound to it, it uses its configured load balancing method to select a service for the request, and forwards that request to the service that it selected.

To configure a basic load balancing setup, you need to perform the following tasks:

- Create load balancing virtual servers
- Create services
- Bind services to the load balancing virtual server

For more information on load balancing, see "Load Balancing." For detailed instructions on setting up a basic load balancing configuration, see "Setting Up Basic Load Balancing."

Configuring a Content Switching Action

Sep 30, 2013

You specify the target load balancing virtual server for a content switching policy when binding the policy to the content switching virtual server. Consequently, you have to configure one policy for each load balancing virtual server to which to direct traffic.

However, if your content switching policy uses a default syntax rule, you can configure an action for the policy. In the action, you can specify the name of the target load balancing virtual server, or you can configure a request-based expression that, at run time, computes the name of the load balancing virtual server to which to send the request. The action expression must be specified in the default syntax.

The expression option can drastically reduce the size of your content switching configuration, because you need only one policy per content switching virtual server. Content switching policies that use an action can also be bound to multiple content switching virtual servers, because the target load balancing virtual server is no longer specified in the content switching policy. The ability to bind a single policy to multiple content switching virtual servers helps to further reduce the size of your content switching configuration.

After you create an action, you create a content switching policy and specify the action in the policy, so that the action is performed when that policy matches a request.

Note: You can also, for a content switching policy that uses a default syntax rule, specify the target load balancing virtual server when binding the policy to a content switching virtual server, instead of using a separate action. For domain-based policies, URL-based policies, and rule based policies that use classic expressions, an action is not available. So, for these types of policies, you specify the name of the target load balancing virtual server when binding the policy to a content switching virtual server. For more information, see "Binding Policies to a Content Switching Virtual Server."

Updated: 2013-08-22

If you choose to specify the name of the target load balancing virtual server in a content switching action, you need as many content switching policies as you have target load balancing virtual servers. Content switching decisions, in this case, are based on the rule in the content switching policy, and the action merely specifies the target load balancing virtual server. When a request matches the policy, the request is forwarded to the specified load balancing virtual server.

To create and verify a content switching action that specifies the name of the target load balancing virtual server, by using the command line interface

At the command prompt, type:

- add cs action <name> -targetLBVserver <string> [-comment <string>]
- show cs action <name>

```
> add cs action mycsaction -targetLBVserver mylbvserver -comment "Forwards requests to mylbvserver."
 Done
> show cs action mycsaction
 Name: mycsaction
 Target LB Vserver: mylbvserver
```

```
Hits: 0
Undef Hits: 0
Action Reference Count: 0
Comment: "Forwards requests to mylbvserver."

Done
>
```

To configure a content switching action that specifies the name of the target load balancing virtual server, by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Actions.
2. In the details pane, do one of the following:
   - To create a content switching action, click Add.
   - To modify a content switching action, select the content switching action, and then click Open.
3. In the Create Content Switching Action or Configure Content Switching Action dialog box, set the following parameters:
   - Name*
   - Target LB Virtual Server* (Under Target LB Virtual Server, click Name.)
   - Comment

   * A required parameter
4. Click Create.

Updated: 2013-08-22

If you choose to configure a request-based expression that can dynamically compute the name of the target load balancing virtual server, you need to configure only one content switching policy to select the appropriate virtual server. The rule for the policy can be a simple TRUE (the policy matches all requests) because, in this case, content switching decisions are based on the expression in the action. By configuring an expression in an action, you can drastically reduce the size of your content switching configuration.

If you choose to configure a request-based expression for computing the name of the target load balancing virtual server at run time, you must carefully consider how to name the load balancing virtual servers in the configuration. You must be able to derive their names by using the request-based policy expression in the action.

For example, if you are switching requests on the basis of the URL suffix (file extension of the requested resource), when naming the load balancing virtual servers, you can follow the convention of appending the URL suffix to a predetermined string, such as mylb_. For example, load balancing virtual servers for HTML pages and PDF files could be named mylb_html and mylb_pdf, respectively. In that case, the rule that you can use in the content switching action, to select the appropriate load balancing virtual server, is "mylb_"+HTTP.REQ.URL.SUFFIX. If the content switching virtual server receives a request for an HTML page, the expression returns mylb_html, and the request is switched to virtual server mylb_html.

To create a content switching action that specifies an expression, by using the command line interface

At the command line, type the following commands to create a content switching action that specifies an expression and verify the configuration:

- add cs action <name> -targetVserverExpr <expression>) [-comment <string>]
- show cs action <name>

```
> add cs action mycsaction1 -targetvserverExpr '"mylb_" + HTTP.REQ.URL.SUFFIX'
 Done
> show cs action mycsaction1
 Name: mycsaction1
 Target Vserver Expression: "mylb_" + HTTP.REQ.URL.SUFFIX
 Target LB Vserver: No_Target
          ...
 Done
>
```

To configure a content switching action that specifies an expression by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Actions.
2. In the details pane, do one of the following:
   - To create a content switching action, click Add.
   - To modify a content switching action, select the content switching action, and then click Open.
3. In the Create Content Switching Action or Configure Content Switching Action dialog box, set the following parameters:
   - Name*
   - Target LB Expression* (Under Target LB Virtual Server, click Expression.)
   - Comment

   * A required parameter
4. Click Create.

Configuring Content Switching Policies

Aug 22, 2013

A content switching policy defines a type of request that is to be directed to a load balancing virtual server. These policies are applied in the order of the priorities assigned to them or (if you are using NetScaler classic policies and do not assign priorities when binding them) in the order in which the policies were created.

The policies can be:

- **Domain-based policies.** The NetScaler appliance compares the domain of an incoming URL with the domains specified in the policies. The appliance then returns the most appropriate content. Domain-based policies must be classic policies; default syntax policies are not supported for this type of content switching policy.
- **URL-based policies.** The appliance compares an incoming URL with the URLs specified in the policies. The appliance then returns the most appropriate URL-based content, which is usually the longest matching configured URL. URL-based policies must be classic policies; default syntax policies are not supported for this type of content switching policy.
- **Rule-based policies.** The appliance compares incoming data to expressions specified in the policies. You create rule-based policies by using either a classic expression or a default syntax expression. Both classic and default syntax policies are supported for rule-based content switching policies.

  Note: A rule based policy can be configured with an optional action. A policy with an action can be bound to multiple virtual servers or policy labels.

  If you set a priority when binding your policies to the content switching virtual server, the policies are evaluated in order of priority. If you do not set specific priorities when binding your policies, the policies are evaluated in the order in which they were created.

For information about NetScaler classic policies and expressions, see "Configuring Classic Policies and Expressions." For information about Default Syntax policies, see "Configuring Default Syntax Expressions."

At the command prompt, type one of the following commands:

- add cs policy <policyName> -domain <domain>
- add cs policy <policyName> -url <URLValue>
- add cs policy <policyName> -rule <RULEValue>
- add cs policy <policyName> -rule <RULEValue> -action <actionName>

Example

```
add cs policy Policy-CS-1 -url "/sports/*"

add cs policy Policy-CS-1 -domain "example.com"

add cs policy Policy-CS-1 -rule "CLIENT.IP.SRC.SUBNET(24).EQ(10.217.84.0)"

add cs policy Policy-CS-2 -rule "SYS.TIME.BETWEEN(GMT 2009 Nov,GMT 2009 Dec)"
```

```
add cs policy Policy-CS-3 -rule "http.req.method.eq(GET)" -action act1
```

At the command prompt, type:

rename cs policy <policyName> <newName>

Example

```
rename cs policy myCSPolicy myCSPolicy1
```

1. In the navigation pane, expand Traffic Management, expand Content Switching, and then click Policies.
2. In the details pane, select the policy you want to rename, and then, in the Action list, click Rename.
3. In the Rename CSW Policy dialog box, in the Name text box, specify a new name for the policy.
4. Click OK.

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, click Add.
3. In the Create Content Switching Policy dialog box, in the Name text box, type the name of the policy (for example, Policy-CS-1).
4. Choose the type of policy that you want to create, and configure the policy.
   - To create a domain-based policy, in the Domain text box, type the domain (for example, example.com).
   - To create a URL-based policy, click URL, and in the Value text box, type an absolute or relative URL (for example, http://www.example.com/sports, or just /sports).
   - To create a rule-based policy, click Configure, and do the following:
     1. In the Create Expression dialog box, choose the expression syntax you want to use.
        - If you want to use default syntax, accept the default and proceed to the next step.
        - If you want to use classic syntax, click Switch to Classic Syntax.

        The Expression portion of the dialog box changes to match your choice. The default syntax Expression view has fewer elements than does the classic syntax Expression view. In the default syntax Expression view, instead of a preview window, a button provides access to an expression evaluator. The evaluator evaluates the expression you entered, to verify that it is valid, and displays an analysis of the expression's effect.
     2. Enter your policy expressions.
        - If you are using classic syntax and need further instructions, see Configuring Classic Policies and Expressions.
        - If you are using the default syntax and need further instructions, see Configuring Default Syntax Expressions.
5. Click Create, and then click Close. The policy you created appears in the Content Switching Policies pane.

Configuring Content Switching Policy Labels

Oct 31, 2013

A policy label is a user-defined bind point to which policies are bound. When a policy label is invoked, all the policies bound to it are evaluated in the order of the priority that you assigned to them. A policy label can include one or more policies, each of which can be assigned its own result. A match on one policy in the policy label can result in proceeding to the next policy, invoking a different policy label or appropriate resource, or an immediate end to policy evaluation and return of control to the policy that invoked the policy label. You can create policy labels for default syntax policies only.

For information about policy labels, see the "Creating Policy Labels."

A content switching policy label consists of a name, a label type, and a list of policies bound to the policy label. The policy label type specifies the protocol that was assigned to the policies bound to the label. It must match the service type of the content switching virtual server to which the policy that invokes the policy label is bound. For example, you can bind TCP Payload policies to a policy label of type TCP only. Binding TCP Payload policies to a policy label of type HTTP is not supported.

Each policy in a content switching policy label is associated with either a target (which is equivalent to the action that is associated with other types of policies, such as rewrite and responder policies) or a gotoPriorityExpression option and/or an invoke option. That is, for a given policy in a content switching policy label, you can specify a target, or you can set the gotoPriorityExpression option and/or the invoke option. Additionally, if multiple policies evaluate to true, only the target of the last policy that evaluates to true is considered.

You can use either the NetScaler command line or the configuration utility to configure content switching policy labels. In the NetScaler command-line interface (CLI), you first create a policy label by using the add cs policylabel command. Then, you bind policies to the policy label, one policy at a time, by using the bind cs policylabel command. In the NetScaler configuration utility, you perform both tasks in a single dialog box.

At the command prompt, type:

add cs policylabel <labelName> <cspolicylabelTypetype>
Example

```
add cs policylabel testpollab http
```

At the command prompt, type:

rename cs policylabel <labelName> <newName>
Example

```
rename cs policylabel oldPolicyLabelName newPolicyLabelName
```

1. In the navigation pane, expand Traffic Management, expand Content Switching, and then click Policy Labels.
2. In the details pane, select the policy label that you want to rename, and then, in the Action list, click Rename.

3. In the Rename CSW Policy Label dialog box, in the Name text box, specify a new name for the policy label.
4. Click OK.

At the command prompt, type the following commands to bind a policy to a policy label and verify the configuration:

- bind cs policylabel <labelName> <policyName> <priority>[[-targetVserver <string>] | [-gotoPriorityExpression <expression>] | [-invoke <labeltype> <labelName>]]
- show cs policylabel <labelName>

Example

```
bind cs policylabel testpollab test_Pol 100 -targetVserver LBVIP
show cs policylabel testpollab
        Label Name: testpollab
        Label Type: HTTP
        Number of bound policies: 1
        Number of times invoked: 0
1)      Policy Name: test_Pol
        Priority: 100
        Target Virtual Server: LBVIP
```

Note: If a policy is configure with an action, the target virtual server (targetVserver), go to priority expression (gotoPriorityExpression), and invoke (invoke) parameters are not required. If a policy is not configure with an action, you need to configure at least one of the following parameters: targetVserver, gotoPriorityExpression, and invoke.

At the command prompt, type the following commands to unbind a policy from a policy label and verify the configuration:

- unbind cs policylabel <labelName> <policyName>
- show cs policylabel <labelName>

Example

```
unbind cs policylabel testpollab test_Pol
show cs policylabel testpollab
        Label Name: testpollab
        Label Type: HTTP
        Number of bound policies: 0
        Number of times invoked: 0
```

At the command prompt, type:

rm cs policylabel <labelName>

1. Navigate to Traffic Management > Content Switching > Policy Labels.
2. In the details pane, do one of the following:
   - To create a new policy label, click Add.

- To modify an existing policy label, select the policy label, and then click Open.

3. In the Create Content Switching Policy Label dialog box, set the following parameters:

   - Name*
   - Label Type*

   * A required parameter

4. To add a policy to a list, click Insert Policy, and then click one of the policies in the drop-down list. If you click New Policy, create a new policy as described in "Creating Content Switching Policies."
5. For the policy that you added to the list, set the following parameters:

   - Priority* (The default value is 100. To modify the value, double-click in the Priority column.)
   - Target
   - Goto Expression
   - Invoke

   * A required parameter

6. To automatically renumber the policies, click Regenerate Priorities.
7. Click Create or OK.

Binding Policies to a Content Switching Virtual Server

Sep 03, 2014

After you create your content switching virtual server and policies, you bind each policy to the content switching virtual server. When binding the policy to the content switching virtual server, you specify the target load balancing virtual server.

Note: If your content switching policy uses a default syntax rule, you can configure a content switching action for the policy. If you configure an action, you must specify the target load balancing virtual server when you are configuring the action, not when you are binding the policy to the content switching virtual server. For more information about configuring a content switching action, see Configuring a Content Switching Action.

At the command prompt, type:

bind cs vserver <name>[-lbvserver<string> -targetLBVServer<string> -policyname <string> -priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type (REQUEST | RESPONSE)] [-invoke (<labelType> <labelName>)]

Example

```
bind cs vserver csw-vip2 -policyname csw-ape-policy2 -priority 14 -gotoPriorityExpression NEXT

bind cs vserver csw-vip3 -policyname rewrite-policy1 -priority 17 -gotoPriorityExpression
'q.header("a").count' -flowtype REQUEST -invoke policylabel label1

bind cs vserver Vserver-CS-1 Vserver-LB-1 -policyname Policy-CS-1 -priority 20
```

Note: The parameters, target load balancing virtual server (targetVserver), go to priority expression (gotoPriorityExpression), and invoke method (invoke) cannot be used if a policy has an action.

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, double-click the virtual server for which you want to bind the policy (for example, Vserver-CS-1).
3. In the Configure Virtual Server (Content Switching) dialog box, on the Policies tab, click Insert Policy. A list of existing policies appears in the Policy Name drop-down list, You can either select an existing policy or create a new policy:
   - Select an existing policy that you previously created to bind to the virtual server.
   - Select New Policy from the Policy Name drop-down list to create a new content switching policy. After you create a new policy, it is automatically bound to the virtual server.
4. Click OK.

Configuring Policy Based Logging for Content Switching

Aug 22, 2013

You can configure policy based logging for a content switching policy. Policy based logging enables you to specify a format for log messages. The contents of the log message are defined by using a default syntax expression in the content switching policy. When the content switching action specified in the policy is performed, the NetScaler appliance constructs the log message from the expression and writes the message to the log file. Policy based logging is particularly useful if you want to test and troubleshoot a configuration in which content switching actions identify the target load balancing virtual server at run time.

Note: If multiple policies bound to a given virtual server evaluate to TRUE and are configured with an audit message action, the NetScaler appliance does not perform all the audit message actions. It performs only the audit message action that is configured for the policy whose content switching action is performed.
To configure policy based logging for a content switching policy, you must first configure an audit message action. For more information about configuring an audit message action, see Configuring Policy-Based Logging. After you configure the audit message action, you specify the action in a content switching policy.

At the command line, type the following commands to configure policy based logging for a content switching policy and verify the configuration:

- set cs policy <policyName> -logAction <string>
- show cs policy <policyName>

Example

```
> set cs policy cspol1 -logAction csLogAction
 Done
> show cs policy cspol1

 Policy: cspol1 Rule: TRUE Action: csact1
 LogAction: csLogAction
 Hits: 0

1) CS Vserver: csvs1
 Priority: 10
 Done
>
```

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, click the policy for which you want to configure policy based logging, and then click Open.
3. In the Configure Content Switching Policy dialog box, from the Log Action list, select the log action that you want.

4. Click OK.

Verifying the Configuration

Mar 15, 2012

To verify that your content switching configuration is correct, you need to view the content switching entities. To verify proper operation after your content switching configuration has been deployed, you can view the statistics that are generated as the servers are accessed.

Updated: 2013-10-31

You can view the properties of content switching virtual servers that you have configured on the NetScaler. You can use the information to verify whether the virtual server is correctly configured and, if necessary, to troubleshoot. In addition to details such as name, IP address, and port, you can view the various policies bound to a virtual server, and its traffic-management settings.

The content switching policies are displayed in the order of their priority. If more than one policy has the same priority, they are shown in the order in which they are bound to the virtual server.

Note: If you have configured the content switching virtual server to forward traffic to a load balancing virtual server, you can also view the content switching policies by viewing the properties of the load balancing virtual server.

To view the properties of content switching virtual servers by using the command line interface

To list basic properties of all content switching virtual servers in your configuration, or detailed properties of a specific content switching virtual server, at the command prompt, type one of the following commands:

- show cs vserver
- show cs vserver <name>

Example

1.

```
show cs vserver Vserver-CS-1
Vserver-CS-1 (10.102.29.161:80) - HTTP Type: CONTENT
State: UP
Last state change was at Thu Jun 30 10:48:59 2011
Time since last state change: 6 days, 20:03:00.760
Client Idle Timeout: 180 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
Appflow logging: DISABLED
Port Rewrite : DISABLED
State Update: DISABLED
Default: Content Precedence: RULE
Vserver IP and Port insertion: OFF
Case Sensitivity: ON
Push: DISABLED Push VServer:
Push Label Rule: none
```

...
1) Policy : __ESNS_PREBODY_POLICY Priority:0
2) Policy : __ESNS_POSTBODY_POLICY Priority:0

1) Compression Policy Name: __ESNS_CMP_POLICY Priority: 2147483647
GotoPriority Expression: END
Flowtype: REQUEST

1) Rewrite Policy Name: __ESNS_REWRITE_POLICY Priority: 2147483647
GotoPriority Expression: END
Flowtype: REQUEST

1) Cache Policy Name: dfbx Priority: 10
GotoPriority Expression: END
Flowtype: REQUEST

1) Responder Policy Name: __ESNS_RESPONDER_POLICY Priority: 2147483647
GotoPriority Expression: END

1) Policy: wiki Target: LBVIP2 Priority: 25 Hits: 0
2) Policy: plain Target: LBVIP1 Priority: 90 Hits: 0
3) Policy: DispOrderTest2 Target: KerbAuthLBVS Priority: 91 Hits: 0
4) Policy: test_Pol Target: LBVIP1 Priority: 92 Hits: 0
5) Policy: PolicyNameTesting Target: LBVIP1 Priority: 100 Hits: 0
Done
>

2.
show cs vserver
1) Vserver-CS-1 (10.102.29.161:80) - HTTP Type: CONTENT
State: UP
...
Appflow logging: DISABLED
Port Rewrite : DISABLED
State Update: DISABLED

2) apubendpt (10.111.111.1:80) - HTTP Type: CONTENT
State: UP
...
Client Idle Timeout: 180 sec
Down state flush: DISABLED
...

3) apubendpt1 (10.111.111.2:80) - HTTP Type: CONTENT
State: UP
...
Disable Primary Vserver On Down : DISABLED

Appflow logging: DISABLED
Port Rewrite : DISABLED
State Update: DISABLED
...

To view the properties of content switching virtual servers by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, click a virtual server to display its configuration details at the bottom of the screen.
3. To display the names of the policies that are bound to the content switching virtual server, double-click the virtual server and then click the Policies tab.

Updated: 2013-08-22

You can view the properties of the content switching policies that you defined, such as the name, domain, and URL or expression, and use the information to find any mistakes in the configuration, or to troubleshoot if something is not working as it should.

To view the properties of content switching policies by using the command line interface

To list either basic properties of all content switching policies in your configuration or detailed properties of a specific content switching policy, at the command prompt, type one of the following commands:

- show cs policy
- show cs policy <PolicyName>

Example

```
show cs policy
```

```
show cs policy Policy-CS-1
```

To view the properties of content switching policies by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, double-click a policy to view the details.
3. To view the policy labels and virtual servers that this policy is bound to, on the Content Switching Policies pane, click Show Bindings.

Updated: 2013-08-22

The Content Switching Visualizer is a tool that you can use to view a content switching configuration in graphical format. You can use the visualizer to view the following configuration items:

- A summary of the load balancing virtual servers to which the content switching virtual server is bound.

- All services and service groups that are bound to the load balancing virtual server and all monitors that are bound to the services.
- The configuration details of any displayed element.
- Any policies bound to the content switching virtual server. These policies need not be content switching policies. Many types of policies, such as Rewrite policies, can be bound to a content switching virtual server.

After you configure the various elements in a content switching and load balancing setup, you can export the entire configuration to an application template file.

Note: The Visualizer requires a graphical interface, so it is available only through the configuration utility.

To view a content switching configuration by using the Visualizer in the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server that you want to view, and then click Visualizer.
3. In the Content Switching Visualizer window, you can adjust the viewable area as follows:
   - Click the Zoom In and Zoom Out icons to increase or decrease the viewable area.
   - Click the Save Image icon to save the graph as an image file.
   - In the Search in text field, begin typing the name of the item you are looking for. When you have typed enough characters to identify the item, its location is highlighted. To restrict the search, click the drop-down menu and select the type of element that you want to search for.
4. To view configuration details for entities that are bound to this virtual server, you can do the following:
   - To view policies that are bound to the virtual server, in the tool bar at the top of the dialog box select one or more feature-specific policy icons. If policy labels are configured, they appear in the main view area.
   - To view the configuration details for a bound service or service group, click the icon for the service, click the Related Tasks tab, and then click Show Member Services.
   - To view the configuration details for a monitor, click the icon for the monitor, click the Related Tasks tab, and then click View Monitor.
5. To view detailed statistics for any virtual server in the content switching configuration, click the virtual server for which you want to view statistics, then click the Related Tasks tab, and then click Statistics.
6. To view a comparative list of the parameters whose values either differ or are not defined across service containers for a load balancing virtual server, click the icon for a container, click the Related Tasks tab, and then click Service Attributes Diff.
7. To view monitor binding details for the services in a container, in the Service Attributes Diff dialog box, in the Group column for the container, click Details. This comparative list helps you determine which service container has the configuration you want to apply to all the service containers.
8. To view the number of requests received per second at a given point in time by the virtual servers in the configuration, and the number of hits per second at a given point in time for rewrite, responder, and cache policies, click Show Stats. The statistical information is displayed on the respective nodes in the Visualizer. This information is not updated in real time. It has to be refreshed manually. To refresh the information, click Refresh Stats.
   Note: This option is available only on NetScaler nCore builds.
9. To copy configuration details for an element to a document or spreadsheet, click the icon for that element, click Related Tasks, click Copy Properties, and then paste the information into a document.
10. To export the entire configuration that is displayed in the Visualizer to an application template file, click the icon for the content switching virtual server, click Related Tasks, and then click Create Template. When creating the application template, you can configure variables in some policy expressions and actions. For more information about creating the application template file and configuring variables for a template, see AppExpert.

Customizing the Basic Content Switching Configuration

Jun 08, 2015

After you configure a basic content switching setup, you might need to customize it to meet your requirements. If your web servers are UNIX-based and rely on case sensitive pathnames, you can configure case sensitivity for policy evaluation. You can also set precedence for evaluation of the content switching policies that you configured. If you want to configure content switching for a specific a virtual LAN, you can configure a content switching virtual server with a listen policy.

To customize the basic content switching configuration, see the following sections:

- Configuring Case Sensitivity for Policy Evaluation
- Setting the Precedence for Policy Evaluation
- Support for Multiple Ports for HTTP and SSL Type Content Switching Virtual Servers
- Configuring per-VLAN Wildcarded Virtual Servers
- Configuring the Microsoft SQL Server Version Setting

Updated: 2013-10-31

You can configure the content switching virtual server to treat URLs as case sensitive in URL-based policies. When case sensitivity is configured, the NetScaler appliance considers case when evaluating policies. For example, if case sensitivity is off, the URLs /a/1.htm and /A/1.HTM are treated as identical. If case sensitivity is on, those URLs are treated as separate and can be switched to different targets.

To configure case sensitivity by using the command line interface

At the command prompt, type:

set cs vserver <name> -caseSensitive (ON | OFF)

Example

```
set cs vserver Vserver-CS-1 -caseSensitive ON
```

To configure case sensitivity by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure case sensitivity (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, select Case Sensitivity check box, and then click OK.

Updated: 2013-10-31

Precedence refers to the order in which policies that are bound to a virtual server are evaluated. You do not normally have to configure precedence: the default precedence works correctly in many cases. If you want to make sure that one policy or set of policies is applied first, however, and another policy or set of policies is applied only if the first set does not match

a request, you can configure either URL-based precedence or rule-based precedence.

Precedence with URL-Based Policies

If there are multiple matching URLs for the incoming request, the precedence (priority) for URL-based policies is:

1. Domain and exact URL
2. Domain, prefix, and suffix
3. Domain and suffix
4. Domain and prefix
5. Domain only
6. Exact URL
7. Prefix and suffix
8. Suffix only
9. Prefix only
10. Default

If you configure precedence based on URL, the request URL is compared to the configured URLs. If none of the configured URLs match the request URL, then rule-based policies are checked. If the request URL does not match any rule-based policies, or if the content group selected for the request is down, then the request is processed as follows:

- If you configure a default group for the content switching virtual server, then the request is forwarded to the default group.
- If the configured default group is down or if no default group is configured, then an "HTTP 404 Not Found" error message is sent to the client.

Note: You should configure URL-based precedence if the content type (for example, images) is the same for all clients. However, if different types of content must be served based on client attributes (such as Accept-Language), you must use rule-based precedence.

Precedence with Rule-Based Policies

If you configure precedence based on rules, which is the default setting, the request is tested on the basis of the rule-based policies you have configured. If the request does not match any rule-based policies, or if the content group selected for the incoming request is down, the request is processed in the following manner:

- If a default group is configured for the content switching virtual server, the request is forwarded to the default group.
- If the configured default group is down or if no default group is configured, an "HTTP 404 Not Found" error message is sent to the client.

At the command prompt, type:

set cs vserver <name> -precedence (RULE | URL)

Example

```
set cs vserver Vserver-CS-1 -precedence RULE
```

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure precedence, (for example, Vserver-CS-1), and

then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, under Precedence, click Rule or URL, and then click OK.

Updated: 2013-10-31

If you want to configure content switching for traffic on a specific virtual local area network (VLAN), you can create a wildcarded virtual server with a listen policy that restricts it to processing traffic only on the specified VLAN.

To configure a wildcarded virtual server that listens to a specific VLAN by using the command line interface

At the command prompt, type:

add cs vserver <name> <serviceType> IPAddress * Port * -listenpolicy <expression> [-listenpriority <positive_integer>]
Example

```
add cs vserver Vserver-CS-vlan1 ANY * *
    -listenpolicy "CLIENT.VLAN.ID.EQ(2)"  -listenpriority 10
```

To configure a wildcarded virtual server that listens to a specific VLAN by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, do one of the following:
   - To create a new virtual server, click Add.
   - To modify an existing virtual server, select the virtual server, and then click Open.
3. In the Create Virtual Server or Configure Virtual Server dialog box, on the Services tab, type or select values for the following parameters:
   - Name*—name
   - Protocol*—type
   - IP address*—IPAddress
   - Port—port

   * A required parameter

4. In the Advanced tab, expand Listen Policy, and then type or select values for the following parameters:
   - Listen Priority*—priority
   - Listen Policy Rule*—rule

   * A required parameter

5. Click Create or OK, depending on whether you are creating a new virtual server or modifying an existing virtual server.
6. Click Close. The virtual server that you created now appears in the Virtual Servers page.
7. To remove a virtual server, in the Virtual Servers pane select the virtual server, and then click Remove.

After you have created this virtual server, you bind it to one or more services as described in Binding Services to the Virtual Server.

Updated: 2013-08-22

You can specify the version of Microsoft® SQL Server® for a content switching virtual server that is of type MSSQL. The version setting is recommended if you expect some clients to not be running the same version as your Microsoft SQL Server product. The version setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

To set the Microsoft SQL Server version parameter by using the command line interface

At the command prompt, type the following commands to set the Microsoft SQL Server version parameter for a content switching virtual server and verify the configuration:

- set cs vserver <name> -mssqlServerVersion <mssqlServerVersion>
- show cs vserver <name>

```
> set cs vserver myMSSQLcsvip -mssqlServerVersion 2008R2
 Done
> show cs vserver myMSSQLcsvip
 myMSSQLcsvip (192.0.2.13:1433) - MSSQL Type: CONTENT
 State: UP
 . . .
 . . .
 MSsql Server Version: 2008R2
 . . .
 . . .
 Done
>
```

To set the Microsoft SQL Server version parameter by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the navigation pane, expand Content Switching, and then click Virtual Servers.
3. In the details pane, select the virtual server for which you want to configure the setting, and then click Open.
4. In the Configure Virtual Server dialog box, do the following:
    1. In the advanced tab, click MsSql.
    2. In the Server Version list, select the version of your Microsoft SQL Server product.
    3. Click Create or OK, and then click Close.

Protecting the Content Switching Setup against Failure

Jun 08, 2015

Content switching may fail when the content switching virtual server goes DOWN or fails to handle excessive traffic, or for other reasons. To reduce the chances of failure, you can take the following measures to protect the content switching setup against failure:

- Configure a backup content switching virtual server
- Configure spillover for preventing the overloading of the primary and diverting excess traffic to the backup virtual server
- Specify a redirect URL, the URL to which the content is switched if both the primary and backup content switching virtual servers are DOWN
- Enable the State Update option for marking a content switching virtual server as DOWN when the load balancing virtual server is DOWN
- Flush the surge queues when the queues become too long

Updated: 2013-11-08

If the primary content switching virtual server is marked DOWN or DISABLED, the NetScaler appliance can direct requests to a backup content switching virtual server. It can also send a notification message to the client regarding the site outage or maintenance. The backup content switching virtual server is a proxy and is transparent to the client.

When configuring the backup virtual server, you can specify the configuration parameter Disable Primary When Down to ensure that, when the primary virtual server comes back up, it remains the secondary until you manually force it to take over as the primary. This is useful if you want to ensure that any updates to the database on the server for the backup are preserved, enabling you to synchronize the databases before restoring the primary virtual server.

You can configure a backup content switching virtual server when you create a content switching virtual server or when you change the optional parameters of an existing content switching virtual server. You can also configure a backup content switching virtual server for an existing backup content switching virtual server, thus creating cascaded backup content switching virtual servers. The maximum depth of cascaded backup content switching virtual servers is 10. The appliance searches for a backup content switching virtual server that is up and accesses that content switching virtual server to deliver the content.

Note: If a content switching virtual server is configured with both a backup content switching virtual server and a redirect URL, the backup content switching virtual server takes precedence over the redirect URL. The redirect is used when the primary and backup virtual servers are down.

To set up a backup content switching virtual server by using the command line interface

At the command prompt, type:

set cs vserver <name> -backupVserver <string> -disablePrimaryOnDown (ON | OFF)

Example

```
set cs vserver Vserver-CS-1 -backupVserver Vserver-CS-2 -disablePrimaryOnDown ON
```

To set up a backup content switching virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to set up a backup content switching virtual server (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, click the Advanced tab.
4. In the Backup Virtual Server list, select the backup virtual server (for example, Vserver-CS-2).
5. If you want to configure the backup server to remain as the primary server after the primary virtual server is brought back up, select the Disable Primary When Down check box.
6. Click OK.

Updated: 2013-11-04

The spillover option diverts new connections arriving at a content switching virtual server to a backup content switching virtual server when the number of connections to the content switching virtual server exceeds the configured threshold value. The threshold value is dynamically calculated, or you can set the value. The number of established connections (in case of TCP) at the virtual server is compared with the threshold value. When the number of connections reaches the threshold, new connections are diverted to the backup content switching virtual server.

If the backup content switching virtual servers reach the configured threshold and are unable to take the load, the primary content switching virtual server diverts all requests to the redirect URL. If a redirect URL is not configured on the primary content switching virtual server, subsequent requests are dropped.

To configure a content switching virtual server to divert new connections to a backup virtual server by using the command line interface

At the command prompt, type:

set cs vserver <name> -soMethod <methodType> -soThreshold <thresholdValue> -soPersistence <persistenceValue> -soPersistenceTimeout <timeoutValue>

Example

```
set cs vserver Vserver-CS-1 -soMethod Connection -soThreshold 1000 -soPersistence enabled -soPersistenceTimeout 2
```

To set a content switching virtual server to divert new connections to a backup virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure spillover (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, under Spillover, in the Method list, select the type of spillover, and in Threshold text box, type the threshold value (for example, Connection and 1000).
4. Select the Persistence check box and, in Persistence Time-out (min) text box, type the timeout value (for example, 2).
5. Click OK.

Updated: 2015-03-19

You can configure a redirect URL to communicate the status of the NetScaler appliance in the event that a content switching virtual server of type HTTP or HTTPS is DOWN or DISABLED. This URL can be local or remote.

Redirect URLs can be absolute URLs or relative URLs. If the configured redirect URL contains an absolute URL, the HTTP redirect is sent to the configured location, regardless of the URL specified in the incoming HTTP request. If the configured redirect URL contains only the domain name (relative URL), the HTTP redirect is sent to a location after appending the incoming URL to the domain configured in the redirect URL.

Citrix recommends using an absolute URL. That is, a URL ending in /, for example www.example.com/ instead of a relative URL. A relative URL redirection might result in the vulnerability scanner reporting a false positive.

Note: If a content switching virtual server is configured with both a backup virtual server and a redirect URL, the backup virtual server takes precedence over the redirect URL. A redirect URL is used when the primary and backup virtual servers are down.
When redirection is configured and the content switching virtual server is unavailable, the appliance issues an HTTP 302 redirect to the user's browser.

To configure a redirect URL for when the content switching virtual server is unavailable by using the command line interface

At the command prompt, type:

set cs vserver <name> -redirectURL <URLValue>

Example

```
set cs vserver Vserver-CS-1 -redirectURL http://www.newdomain.com/mysite/maintenance
```

To configure a redirect URL for when the content switching virtual server is unavailable by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure a redirect URL (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, in the Redirect URL text box, type the redirect URL (for example, http://www.newdomain.com/mysite/maintenance).
4. Click OK.

Updated: 2014-12-12

The content switching feature enables the distribution of client requests across multiple servers on the basis of the specific content presented to the users. For efficient content switching, the content switching virtual server distributes the traffic to the load balancing virtual servers according to the

content type, and the load balancing virtual servers distribute the traffic to the physical servers according to the specified load balancing method.

For smooth traffic management, it is important for the content switching virtual server to know the status of the load balancing virtual servers. The state update option helps to mark the content switching virtual server DOWN if the load balancing virtual server bound to it is marked DOWN. A load balancing virtual server is marked DOWN if all the physical servers bound to it are marked DOWN.

When State Update is disabled:

The status of the content switching virtual server is marked as UP. It remains UP even if there is no bound load balancing virtual server that is UP.

When State Update is enabled:

When you add a new content switching virtual server, initially, its status is shown as DOWN. When you bind a load balancing virtual server whose status is UP, the status of the content switching virtual server becomes UP.

If more than one load balancing virtual server is bound and if one of them is specified as the default, the status of the content switching virtual server reflects the status of the default load balancing virtual server.

If more than one load balancing virtual server is bound without any of them being specified as the default, the status of the content switching virtual server is marked UP only if all the bound load balancing virtual servers are UP.

To configure the state update option by using the command line interface

At the command prompt, type:

add cs vserver <name> <protocol> <ipAddress> <port> -stateUpdate ENABLED

add cs vserver csw_vserver HTTP 10.18.250.154 80 -stateupdate ENABLED -cltTimeout 180

To configure the state update option by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, do one of the following:
   - To add a virtual server, click Add.
   - To modify a virtual server, select the server, and click Open.
3. In the Create Virtual Server (Content Switching) dialog box, specify values for the following parameters, which correspond to parameters described in "Parameters for configuring content switching" as shown:
   - Name-vServerName
   - IP Address-ipAddress
     Note: If you need to enter an IPv6 address, select the IPv6 check box before you enter the address.
   - Port-port
   - Protocol-protocol
4. On the Advanced tab, select the State Update check box.
5. Click Create.
6. Select the new virtual server, click Open, and verify the settings.

Updated: 2013-12-04

When a physical server receives a surge of requests, it becomes slow to respond to the clients that are currently connected to it, which leaves users dissatisfied and disgruntled. Often, the overload also causes clients to receive error pages. To avoid such overloads, the NetScaler appliance provides features such as surge protection, which controls the rate at which new connections to a service can be established.
The appliance does connection multiplexing between clients and physical servers. When it receives a client request to access a service on a server, the appliance looks for an already established connection to the server that is free. If it finds a free connection, it uses that connection to establish a virtual link between the client and the server. If it does not find an existing free connection, the appliance establishes a new connection with the server, and establishes a virtual link between client and the server. However, if the appliance cannot establish a new connection with the server, it sends the client request to a surge queue. If all the physical servers bound to the load balancing or content switching virtual server reach the upper limit on client connections (max client value, surge protection threshold or maximum capacity of the service), the appliance cannot establish a connection with any server. The surge protection feature uses the surge queue to regulate the speed at which connections are opened with the physical servers. The appliance maintains a different surge queue for each service bound to the virtual server.

The length of a surge queue increases whenever a request comes for which the appliance cannot establish a connection, and the length decreases

whenever a request in the queue gets sent to the server or a request gets timed out and is removed from the queue.

If the surge queue for a service or service group becomes too long, you may want to flush it. You can flush the surge queue of a specific service or service group, or of all the services and service groups bound to a load balancing virtual server. Flushing a surge queue does not affect the existing connections. Only the requests present in the surge queue get deleted. For those requests, the client has to make a fresh request.

You can also flush the surge queue of a content switching virtual server. If a content switching virtual server forwards some requests to a particular load balancing virtual server, and the load balancing virtual server also receives some other requests, when you flush the surge queue of the content switching virtual server, only the requests received from this content switching virtual server are flushed; the other requests in the surge queue of the load balancing virtual server are not flushed.

Note: You cannot flush the surge queues of cache redirection, authentication, VPN or GSLB virtual servers or GSLB services.
Note: Do not use the Surge Protection feature if Use Source IP (USIP) is enabled.

To flush a surge queue by using the command line interface

The flush ns surgeQ command works in the following manner:

- You can specify the name of a service, service group, or virtual server whose surge queue has to be flushed.
- If you specify a name while executing the command, surge queue of the specified entity will be flushed. If more than one entity has the same name, the appliance flushes surge queues of all those entities.
- If you specify the name of a service group, and a server name and port while executing the command, the appliance flushes the surge queue of only the specified service group member.
- You cannot directly specify a service group member (<serverName> and <port>) without specifying the name of the service group (<name>) and you cannot specify <port> without a <serverName>. Specify the <serverName> and <port> if you want to flush the surge queue for a specific service group member.
- If you execute the command without specifying any names, the appliance flushes the surge queues of all the entities present on the appliance.
- If a service group member is identified with a server name, you must specify the server name in this command; you cannot specify its IP address.

At the command prompt, type:

flush ns surgeQ [-name <name>] [-serverName <serverName> <port>]

1.
flush ns surgeQ –name SVC1ANZGB –serverName 10.10.10.1 80
The above command flushes the surge queue of the service or virtual server that is named SVC1ANZGB and has IP address as 10.10.10
2.
flush ns surgeQ
The above command flushes all the surge queues on the appliance.

To flush a surge queue by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. To select an entity, do one of the following:
   - To flush the surge queue of a virtual server, click Virtual Servers, and then select the virtual server.
   - To flush the surge queue of a service, click Services, and then select the service.
   - To flush the surge queue of all the members in a service group, click Service Groups, and then select the service group.
   - To flush the surge queue of a specific member in a service group, click Service Groups, and in the action pane, click Manage Members. In the Manage Members of a Service Group dialog box, select the service group member.

   Note: You can select multiple entities in any window.
   Note: To flush the surge queue of a content switching virtual server, in Steps 1 and 2, expand Content Switching, and then select a virtual server.
3. In the action pane, click Flush Surge Queue.
4. Click OK.
   Note: On the appliance, if there are other entities with the same name as you selected, you are alerted that the surge queues of those entities would also be flushed. Take an appropriate action.

Managing a Content Switching Setup

Jun 08, 2015

After a content switching setup is configured, it may require periodic changes. When operating systems or software are updated, or hardware wears out and is replaced, you may need to take down your setup. Load on your setup may increase, requiring additional resources. You may also modify the configuration to improve performance.

These tasks may require unbinding policies from the content switching virtual server, or disabling or removing content switching virtual servers. After you have made changes to your setup, you may need to re-enable servers and rebind policies. You might also want to rename your virtual servers.

To manage a content switching setup, see the following sections:

- Unbinding Policies from the Content Switching Virtual Server
- Removing Content Switching Virtual Servers
- Disabling and Re-Enabling Content Switching Virtual Servers
- Renaming Content Switching Virtual Servers
- Managing Content Switching Policies
- Modifying a Content Switching Configuration by Using the Visualizer

Updated: 2014-09-03

When you unbind a content switching policy from its virtual server, the virtual server no longer includes that policy when determining where to direct requests.

To unbind a policy from a content switching virtual server by using the command line interface

At the command prompt, type:

unbind cs vserver <name> -policyname <string>

Example

```
unbind cs vserver Vserver-CS-1 -policyname Policy-CS-1
```

To unbind a policy from a content switching virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server form which you want to unbind the policy (for example, Vserver-CS-1), and click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Policies tab, in the Active column, clear the check box next to the policy that you want to unbind from the virtual server (for example, Policy-CS-1).
4. Click OK.

Updated: 2013-10-31

You normally remove a content switching virtual server only when you no longer require the virtual server. When you remove a content switching virtual server, the NetScaler appliance first unbinds all policies from the content switching virtual server, and then removes it.

To remove a content switching virtual server by using the command line interface

At the command prompt, type:

rm cs vserver <name>@
Example

```
rm cs vserver Vserver-CS-1
```

To remove a content switching virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server that you want to remove (for example, Vserver-CS-1), and then click Remove.
3. In the Remove dialog box, click Yes.

Updated: 2013-10-31

Content switching virtual servers are enabled by default when you create them. You can disable a content switching virtual server for maintenance. If you disable the content switching virtual server, the state of the content switching virtual server changes to Out of Service. While out of service, the content switching virtual server does not respond to requests.

To disable or re-enable a virtual server by using the command line interface

At the command prompt, type one of the following commands:

- disable cs vserver <name>@
- enable cs vserver <name>@

Example

```
disable cs vserver Vserver-CS-1
```

```
enable cs vserver Vserver-CS-1
```

To disable or re-enable a virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server that you want to disable (for example, Vserver-CS-1).
3. Disable or re-enable the virtual server by clicking Disable or Enable, and then clicking Yes to confirm your choice.

Updated: 2013-10-31

You can rename a content switching virtual server without unbinding it. The new name is propagated automatically to all affected parts of the NetScaler configuration.

To rename a virtual server by using the command line interface

At the command prompt, type:

rename cs vserver <name>@ <newName>@
Example

rename cs vserver Vserver-CS-1 Vserver-CS-2

To rename a virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server that you want to rename (for example, Vserver-CS-1).
3. Click Rename.
4. In the Name text box, type a new name for the virtual server.
5. Click OK to save your changes.

Updated: 2013-08-22

You can modify an existing policy by configuring rules or changing the URL of the policy, or you can remove a policy. You can also rename an existing advanced content switching policy. You can create different policies based on the URL. URL-based policies can be of different types, as described in the following table.

Table 1. Examples of URL-Based Policies

Type of URL-Based Policy	Specifies
Domain and Exact URL	Requests must match the configured domain name and configured URL (an exact prefix match if only the prefix is configured; or an exact match of the prefix and suffix if both the prefix and suffix are configured). **Example:** **add cs policy Policy-CS-1 -url /sports/tennis/index.html -domain "www.domainxyz.com"**
Domain and Wild Card URL	Requests must match the exact domain name and a partial prefix of the configured URL. **Example:** **add cs policy Policy-CS-1 -url /*.jsp -domain "www.domainxyz.com"**
Domain Only	Requests need match only the configured domain name. **Example:** **add cs policy Policy-CS-1 -domain "www.domainxyz.com"**

Type of URL-Based Policy	Specifies
The Exact URL	The incoming URL must exactly match the URL specified by the policy. If only a URL prefix rule is configured, there must be an exact prefix match with the incoming URL. If a URL prefix and suffix-based rule is configured, there should be an exact match of the prefix and suffix with the incoming URL. **Example:** **add cs policy Policy-CS-1 -url /sports/tennis/index.html**
Prefix Only (Wild Card URL)	All the incoming URLs must start with the configured prefix. **Example:** **add cs policy Policy-CS-1 -url /sports*** sports/*" matches all URLs under /sports "/sports*" matches all URLs whose prefix match "/sports" starting from the beginning of a URL
Suffix Only (Wild Card URL)	All incoming URLs must end with the configured URL suffix. **Example:** **add cs policy Policy-CS-1 -url /*.jsp** "/*.jsp" matches all URLs whose file extension is "jsp"
Prefix and Suffix (Wild Card URL)	All incoming URLs must start with the configured prefix and end with the configured suffix. **Example:** **add cs policy Policy-CS-1 -url /sports/*.jsp**

Note: You can configure rule-based content switching using classical policy expressions or advanced policy expressions.

To modify, remove, or rename a policy by using the command line interface

At the command prompt, type one of the following commands:

- set cs policy <policyName> [-domain <domainValue>] [-rule <ruleValue>] [-url <URLValue>]
- rm cs policy <policyName>
- rename cs policy <policyName> <newPolicyName>

Example

```
set cs policy Policy-CS-1 -domain "www.domainxyz.com"

set cs policy Policy-CS-1 -rule "CLIENT.IP.SRC.SUBNET(22).EQ(10.100.148.0)"

set cs policy Policy-CS-2 -rule "SYS.TIME.BETWEEN(GMT 2010 Jun,GMT 2010 Jul)"
```

```
set cs policy Policy-CS-1 -url /sports/*

rename cs policy Policy-CS-1 Policy-CS-11

rm cs policy Policy-CS-1
```

To modify, remove, or rename a policy by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, select the policy that you want to modify (for example, Policy-CS-1).
3. Modify, remove, or rename the policy.
   - To modify the policy, click Open and then make the changes that you want. For example, you can type a new domain name in the Domain text box. Then, click Yes to confirm your changes.
   - To remove the policy, click Remove, and then click Yes to confirm your choice.
   - To rename the policy, click Rename, and specify the new name in the Name text field in the Rename CSW Policy dialog box, and then click OK.
4. In the Configure Content Switching Policies dialog box, in the Domain text box, type the domain name (for example, www.domainxyz.com).
5. Click OK.

Updated: 2013-08-22

You can use the Visualizer to modify a load balancing virtual server to which the content switching virtual server is bound. You can also modify a service or group of similar services, or a monitor. For more information, see "The Load Balancing Visualizer."

Managing Client Connections

May 21, 2015

To ensure efficient management of client connections, you can configure the content switching virtual servers on the NetScaler appliance to use the following features:

- Redirecting client requests to a cache
- Enabling delayed cleanup of virtual server connections
- Rewriting ports and protocols for redirection
- Inserting the IP address and port of a virtual server in the request header
- Setting a time-out value for idle client connections
- Identifying Connections with the 4-tuple and Layer 2 Connection Parameters
- Configuring the ICMP Response. You can configure the NetScaler to send ICMP responses to PING requests according to your settings. On the IP address corresponding to the virtual server, set the ICMP RESPONSE to VSVR_CNTRLD, and on the virtual server, set the ICMP VSERVER RESPONSE.
  The following settings can be made on a virtual server:
  - When you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.
  - When you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds even if one virtual server is UP.
  - When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds even if one virtual server set to ACTIVE is UP.

Updated: 2013-10-31

The NetScaler cache redirection feature redirects HTTP requests to a cache. You can significantly reduce the burden of responding to HTTP requests and improve your Web site performance through proper implementation of the cache redirection feature.

A cache stores frequently requested HTTP content. When you configure cache redirection on a virtual server, the NetScaler appliance sends cacheable HTTP requests to the cache and non-cacheable HTTP requests to the origin Web server. For more information on cache redirection, see "Cache Redirection."

To configure cache redirection on a virtual server by using the command line interface

At the command prompt, type:

set cs vserver <name> -cacheable <Value>

Example

```
set cs vserver Vserver-CS-1 -cacheable yes
```

To configure cache redirection on a virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure cache redirection (for example, Vserver-CS-1), and then click Open.

3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, select the Cache Redirection check box.
4. Click OK.

Updated: 2013-10-31

Under certain conditions, you can configure the down state flush setting to terminate existing connections when a service or a virtual server is marked DOWN. Terminating existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups.

To configure the down state flush setting on a virtual server by using the command line interface

At the command prompt, type:

set cs vserver <name> -downStateFlush <Value>
Example

```
set cs vserver Vserver-CS-1 -downStateFlush enabled
```

To configure the down state flush setting on a virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure down state flush (for example, Vserver-CS-1), and click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, click the Advanced tab.
4. Select the Down state flush check box, and then click OK.

Updated: 2013-10-31

Virtual servers and the services that are bound to them may use different ports. When a service responds to an HTTP connection with a redirect, you may need to configure the NetScaler appliance to modify the port and the protocol to ensure that the redirection goes through successfully. You do this by enabling and configuring the redirectPortRewrite setting.

To configure HTTP redirection on a virtual server by using the command line interface

At the command prompt, type:

set cs vserver <name> -redirectPortRewrite <Value>
Example

```
set cs vserver Vserver-CS-1 -redirectPortRewrite enabled
```

To configure HTTP redirection on a virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure HTTP redirection (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, click the Advanced tab.
4. Select the Redirect Port Rewrite check box, and then click OK.

Updated: 2013-10-31

If you have multiple virtual servers that communicate with different applications on the same service, you must configure the NetScaler appliance to add the IP address and port number of the appropriate virtual server to the HTTP requests that are sent to that service. This setting allows applications running on the service to identify the virtual server that sent the request.

If the primary virtual server is down and the backup virtual server is up, the configuration settings of the backup virtual server are added to the client requests. If you want the same header tag to be added, regardless of whether the requests are from the primary virtual server or backup virtual server, you must configure the required header tag on both virtual servers.

Note: This option is not supported for wildcarded virtual servers or dummy virtual servers.

To insert the IP address and port of the virtual server in the client requests by using the command line interface

At the command prompt, type:

set cs vserver <name> -insertVserverIPPort <vServerIPPORT>
Example

```
set cs vserver Vserver-CS-1 -insertVserverIPPort 10.201.25.136:80
```

To insert the IP address and port of the virtual server in the client requests by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure virtual server port insertion (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, click the Advanced tab.
4. In the Vserver IP Port Insertion list, select the VIPADDR or V6TOV4MAPPING, and then type the port header in a text box next to Vserver IP Port Insertion box.
5. Click OK.

Updated: 2013-10-31

You can configure a virtual server to terminate any idle client connections after a configured time-out period elapses. When you configure this setting, the NetScaler appliance waits for the time you specify and, if the client is idle after that time, it closes the client connection.

To set a time-out value for idle client connections by using the command line

interface

At the command prompt, type:

set cs vserver <name> -cltTimeout <Value>
Example

set cs vserver Vserver-CS-1 -cltTimeout 100

To set a time-out value for idle client connections by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server for which you want to set a time-out value (for example, Vserver-CS-1), and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, click the Advanced tab.
4. In the Client Time-out (secs) text box, type the time-out value (for example, 100).
5. Click OK.

Updated: 2013-08-22

You can now set the L2Conn option for a content switching virtual server. With the L2Conn option set, connections to the content switching virtual server are identified by the combination of the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>) and Layer 2 connection parameters. The Layer 2 connection parameters are the MAC address, VLAN ID, and channel ID.

To set the L2Conn option for a content switching virtual server by using the command line interface

At the command line, type the following commands to configure the L2Conn parameter for a content switching virtual server and verify the configuration:

- set cs vserver <name> -l2Conn (**ON** | **OFF**)
- show cs vserver <name>

```
> set cs vserver mycsvserver -l2Conn ON
Done
> show cs vserver mycsvserver
    mycsvserver (192.0.2.56:80) - HTTP   Type: CONTENT
    State: UP
       . . .
       . . .
    L2Conn: ON Case Sensitivity: ON
       . . .
       . . .
Done
>
```

To set the L2Conn option for a content switching virtual server by using the

configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, click the content switching virtual server for which you want to set the L2Conn option, and then click Open.
3. In the Configure Virtual Server (Content Switching) dialog box, on the Advanced tab, select L2 Connection.
4. Click OK.

Troubleshooting

Jul 22, 2013

If the content switching feature does not work as expected after you have configured it, you can use some common tools to access NetScaler resources and diagnose the problem.

Updated: 2013-07-22

For best results, use the following resources to troubleshoot a content switching issue on a NetScaler appliance:

- Configuration file
- Relevant newnslog file
- Trace files
- Network topology diagram for the network setup of the customer
- Citrix documentation, such as release notes, Knowledge Center articles, and eDocs

In addition to the above resources, the following tools expedite troubleshooting:

- The iehttpheaders or a similar utility
- The Wireshark application customized for the NetScaler trace files
- An SSH utility for command line access
- A HyperTerminal utility to access the console

Updated: 2013-08-02

The most common content switching issues involve the content switching feature not working at all, or working only intermittently, and Service Unavailable responses.

- **Issue**
  The content switching feature is not functioning.

  Resolution

  Check the configuration as follows:
  - Verify that the appliance is licensed for content switching.
  - Verify that the feature is enabled.
  - From the configuration file, verify that valid content switching policies are correctly bound to the load balancing virtual servers.
- **Issue**
  Client receives a 503 - Service Unavailable response.

  Resolution
  - Verify the URL and policy bindings. The client receives the 503 response when none of the policies you have configured is evaluated and no default load balancing virtual server is defined and bound to the content switching virtual server.
  - From the configuration, verify the policies and URL being accessed by the client.
  - Verify that for every type of request the respective policy is evaluated. If the policy is not evaluated, check the policy

expression and update it if necessary.
- Verify the URL and HTTP request and response headers. To do so, record an HTTPHeader trace and, if necessary, record the packet traces on the appliance and the client.

- **Issue**
  Intermittently, the content switching feature is not working as expected.

  Resolution
  - Study the network topology diagram, if available, of the setup to understand the various devices installed between the client and the server(s).
  - Verify the configuration and policy bindings. Make sure that the URL in the policy expression matches to the one in the client request.
  - Verify that appropriate priorities are assigned to the policies. An incorrect precedence or priority assigned to a policy can cause a problem.
  - Run the following commands to verify the bindings and the values of the policy hit counters in the output of the commands:
    ```
    show cs vserver <CS VServer>

    show cs policy <CS Policy>

    stat cs vserver <CS VServer>
    ```
  - Using iehttpheaders or a similar utility, determine whether the HTTP headers for the requests or responses provide any pointers to the issue.
  - Check the release notes and Knowledge Center articles.
  - If the issue is still not resolved, contact Citrix Technical Support with appropriate data for further investigation.

DataStream

May 25, 2015

The NetScaler DataStream feature provides an intelligent mechanism for request switching at the database layer by distributing requests based on the SQL query being sent.

When deployed in front of database servers, a NetScaler ensures optimal distribution of traffic from the application servers and Web servers. Administrators can segment traffic according to information in the SQL query and on the basis of database names, usernames, character sets, and packet size.

You can either configure load balancing to switch requests based on load balancing algorithms or elaborate the switching criteria by configuring content switching to make a decision based on an SQL query parameters. You can further configure monitors to track the state of database servers.

Note: NetScaler DataStream is supported only for MySQL and MS SQL databases. For information about the supported protocol version, character sets, special queries, and transactions, see DataStream Reference.

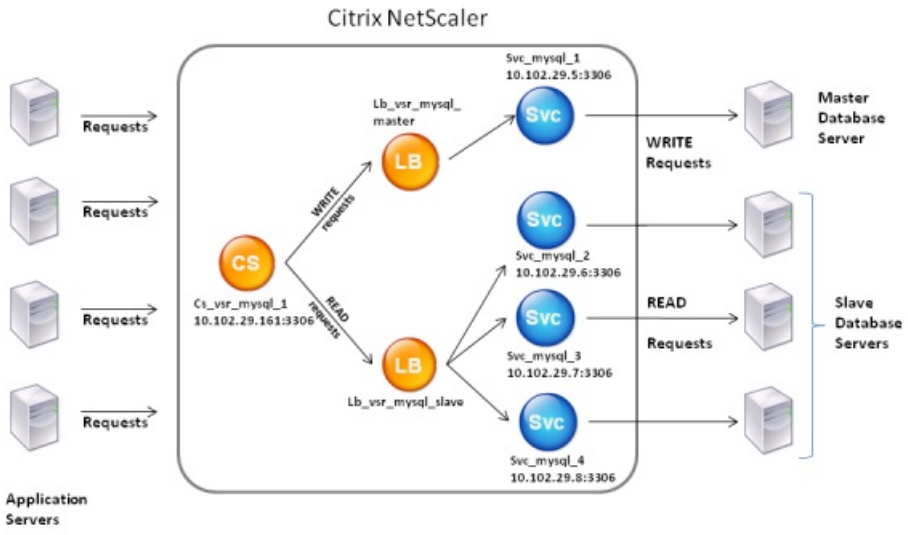
How NetScaler DataStream Works

Updated: 2013-09-18

In DataStream, the NetScaler is placed in-line between the application and/or Web servers and the database servers. On the NetScaler appliance, the database servers are represented by services.

A typical DataStream deployment consists of the entities described in the following diagram.

Figure 1. *DataStream Entity Model*



As shown in this figure, a DataStream configuration can consist of an optional content switching virtual server (CS), a load balancing setup consisting of load balancing virtual servers (LB1 and LB2) and services (Svc1, Svc2, Svc3, and Svc4), and content switching policies (optional).

The clients (application or Web servers) send requests to the IP address of a content switching virtual server (CS) configured on the NetScaler appliance. The NetScaler, then, authenticates the clients using the database user credentials configured on the NetScaler appliance. The content switching virtual server (CS) applies the associated content switching policies to the requests. After evaluating the policies, the content switching virtual server (CS) routes the requests to the appropriate load balancing virtual server (LB1 or LB2), which, then, distributes the requests to the appropriate database servers (represented by services on the NetScaler) based on the load balancing algorithm. The NetScaler uses the same database user credentials to authenticate the connection with the database server.

If a content switching virtual server is not configured on the NetScaler, the clients (application or Web servers) send their requests to the IP address of a load balancing virtual server configured on the NetScaler appliance. The NetScaler authenticates the client by using the database user credentials configured on the NetScaler appliance, and then uses the same credentials to authenticate the connection with the database server. The load balancing virtual server distributes the requests to the database servers according to the load balancing algorithm. The most effective load balancing algorithm for database switching is the least connection method.

DataStream uses connection multiplexing to enable multiple client-side requests to be made over the same server-side connection. The following connection properties are considered:

- User name
- Database name
- Packet size
- Character set

Configuring Database Users

Oct 21, 2014

In databases, a connection is always stateful, which means that as soon as a connection is established, it must be authenticated.

You need to configure your database user name and password on the NetScaler ADC. For example, if you have a user John configured on the database, you need to configure the user John on the ADC too. When you add the database user names and passwords on the ADC, these are added to the nsconfig file.
Note: Names are case sensitive.
The ADC uses these user credentials to authenticate the clients, and then authenticate the server connections with the database servers.

To add a database user by using the command line interface

At the command prompt, type

add db user <username> - password <password>
Example

```
> add db user nsdbuser -password dd260427edf
```

To add a database user by using the configuration utility

1. In the details pane, click Add.
2. In the Create Database User dialog box, specify values for the following parameters.
   - User Name
   - Password
   - Confirm Password
3. Click Create, and then click Close. The user you created appears in the Database Users pane.

If you have changed the password of the database user on the database server, you must reset the password of the corresponding user configured on the NetScaler.

To add a database user by using the configuration utility

1. Navigate to System > User Administration > Database Users, and then click Add.

If you have changed the password of the database user on the database server, you must reset the password of the corresponding user configured on the NetScaler.

To reset the password of a database user by using the command line interface

At the command prompt, type

set db user <username> -password <password>
Example

```
> set db user nsdbuser -password dd260538abs
```

To reset the password of database users by using the configuration utility

1. Navigate to System > User Administration > Database Users.
2. In the details pane, select the database user for which you want to reset the password, and then click Open.
3. In the Configure Database User dialog box, modify the values for the following parameters.
   - Password
   - Confirm Password
4. Click OK.

If a database user no longer exists on the database server, you can remove the user from the NetScaler. However, if the user continues to exist on the database server and you remove the user from the NetScaler, any request from the client with this user name does not get authenticated, and therefore, does not get routed to the database server.

To remove a database user by using the command line interface

At the command prompt, type

rm db user <username>
Example

```
>  rm db user nsdbuser
```

To remove a database user by using the configuration utility

1. Navigate to System > User Administration > Database Users.
2. In the details pane, select the database user that you want to remove, and then click Open.
3. In the Proceed message box, click Yes.

Configuring a Database Profile

Sep 03, 2014

A database profile is a named collection of parameters that is configured once but applied to multiple virtual servers that require those particular parameter settings. After creating a database profile, you bind it to load balancing or content switching virtual servers. You can create as many profiles as you need.

To create a database profile by using the command line interface

At the command line, type the following commands to create a database profile and verify the configuration:

- add db dbProfile <name> [-interpretQuery (YES | NO)] [-stickiness (YES | NO)] [-kcdAccount <string>]
- show db dbProfile

Example

```
> add dbProfile myDBProfile -interpretQuery YES -stickiness YES -kcdAccount mykcdaccnt
 Done
> show dbProfile myDBProfile
 Name: myDBProfile
 Interpret Query: YES
 Stickyness: YES
 KCD Account: mykcdaccnt
 Reference count: 0

 Done
>
```

To create a database profile by using the configuration utility

1. Navigate to System > Profiles.
2. In the details pane, on the Database Profiles tab, do one of the following:
   - To create a database profile, click Add.
   - To modify a database profile, click the profile, and then click Open.
3. In the Create Database Profile or Configure Database Profile dialog box, set the following parameters:
   - Name*
   - KCD Account
   - Interpret Query
   - Stickiness

   * Required parameter
4. Click Create or OK, and then click Close.

To bind a database profile to a load balancing or content switching virtual server by using the command line interface

At the command line, type:

set (lb | cs) vserver <name> -dbProfileName <string>

To bind a database profile to a load balancing or content switching virtual server by using the configuration utility

1. In the navigation pane, expand Traffic Management, and then expand Load Balancing or Content Switching, depending on the type of virtual server to which you want to bind the database profile.
2. Click Virtual Servers.
3. In the details pane, select the virtual server, and then click Open.
4. In the Configure Virtual Server (Load Balancing) or Configure Virtual Server (Content Switching) dialog box, on the Profiles tab, in the Database Profile list, select the database profile.
5. Click OK.

Configuring Load Balancing for DataStream

May 15, 2015

Before configuring a load balancing setup, you must enable the load balancing feature. Then, begin by creating at least one service for each database server in the load balancing group. With the services configured, you are ready to create a load balancing virtual server and bind the services to the virtual server.

Parameter values specific to DataStream

Protocol

Use the MYSQL protocol type for MySQL databases and MSSQL protocol type for MS SQL databases while configuring virtual servers and services. The MySQL and TDS protocols are used by the clients to communicate with the respective database servers by using SQL queries. For information about the MySQL protocol, see http://dev.mysql.com/doc/internals/en/client-server-protocol.html. For information about the TDS protocol, see http://msdn.microsoft.com/en-us/library/dd304523(v=prot.13).aspx.

Port

Port on which the virtual server listens for client connections. Use port 3306 for MySQL database servers.

Method

It is recommended that you use the Least Connection method for better load balancing and lower server load. However, other methods, such as Round Robin, Least Response Time, Source IP Hash, Source IP Destination IP Hash, Least Bandwidth, Least Packets, and Source IP Source Port Hash, are also supported.

Note: URL Hash method is not supported for DataStream.

MS SQL Server Version

If you are using the Microsoft SQL Server, and you expect some clients to not be running the same version as your Microsoft SQL Server product, set the Server Version parameter for the load balancing virtual server. The version setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version. For more information about setting the Server Version parameter, see Configuring the MySQL and Microsoft SQL Server Version Setting.

MySQL Server Version

If you are using the MySQL Server, and you expect some clients to not be running the same version as your MySQL Server product, set the Server Version parameter for the load balancing virtual server. The version setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version. For more information about setting the Server Version parameter, see Configuring the MySQL and Microsoft SQL Server Version Setting.

Configuring Content Switching for DataStream

May 15, 2015

You can segment traffic according to information in the SQL query, on the basis of database names, user names, character sets, and packet size.

You can configure content switching policies with default syntax expressions to switch content based on connection properties, such as user name and database name, command parameters, and the SQL query to select the server.

The default syntax expressions evaluate traffic associated with MYSQL and MS SQL database servers. You can use request-based expressions in default syntax policies to make request switching decisions at the content switching virtual server bind point and response-based expressions (expressions that begin with MYSQL.RES) to evaluate server responses to user-configured health monitors.

Note: For information about default syntax expressions, see Default Syntax Expressions: DataStream.

Parameter values specific to DataStream

Protocol

Use the MYSQL protocol type for MySQL databases and MSSQL protocol type for MS SQL databases while configuring virtual servers and services. The MySQL and TDS protocols are used by the clients to communicate with the respective database servers by using SQL queries. For information about the MySQL protocol, see http://dev.mysql.com/doc/internals/en/client-server-protocol.html. For information about the TDS protocol, see http://msdn.microsoft.com/en-us/library/dd304523(v=prot.13).aspx.

Port

Port on which the virtual server listens for client connections. Use port 3306 for MySQL database servers.

MS SQL Server Version

If you are using Microsoft SQL Server, and you expect some clients to not be running the same version as your Microsoft SQL Server product, set the Server Version parameter for the content switching virtual server. The version setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version. For more information about setting the Server Version parameter, see Configuring the Microsoft SQL Server Version Setting.

Configuring Monitors for DataStream

Oct 01, 2013

To track the state of each load balanced database server in real time, you need to bind a monitor to each service. The monitor is configured to test the service by sending periodic probes to the service. (This is sometimes referred to as performing a health check.) If the monitor receives a timely response to its probes, it marks the service as UP. If it does not receive a timely response to the designated number of probes, it marks the service as DOWN.

For DataStream, you need to use the built-in monitors, MYSQL-ECV and MSSQL-ECV. This monitor provides the ability to send an SQL request and parse the response for a string.

Before configuring monitors for DataStream, you must add database user credentials to your NetScaler. For information about configuring monitors, see Monitors.

When you create a monitor, a TCP connection is established with the database server, and the connection is authenticated by using the user name provided while creating the monitor. You can then run an SQL query to the database server and evaluate the server response to check whether it matches the configured rule.

Examples

In the following example, the value of the error message is evaluated to determine the state of the server.

```
add lb monitor lb_mon1 MYSQL_ECV -sqlQuery "select * from
table2;" -evalrule "mysql.res.error.message.contains(\"Invalid
User\")"-database "NS" -userName "user1"
```

In the following example, the number of rows in the response is evaluated to determine the state of the server.

```
add lb monitor lb_mon4 MYSQL_ECV -sqlQuery "select * from
table4;" -evalrule "mysql.res.atleast_rows_count(7)" -database "NS" -userName "user2"
```

In the following example, the value of a particular column is evaluated to determine the state of the server.

```
add lb monitor lb_mon3 MYSQL_ECV
-sqlQuery "select * from ABC;" -evalrule "mysql.res.row(1).double_elem(2) == 345.12"
-database "NS" -userName "user3"
```

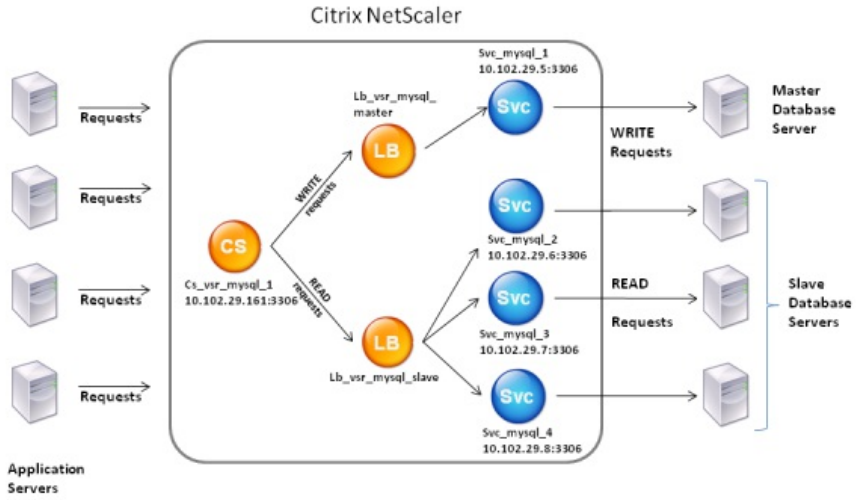
Use Case 1: Configuring DataStream for a Master/Slave Database Architecture

May 15, 2015

A commonly used deployment scenario is the master/slave database architecture where the master database replicates all information to the slave databases.
For master/slave database architecture, you may want all WRITE requests to be sent to the master database and all READ requests to the slave databases.

The following figure shows the entities and the values of the parameters you need to configure on the appliance.

Figure 1. *DataStream Entity Model for Master/Slave Database Setup*



In this example scenario, a service (Svc_mysql_1) is created to represent the master database and is bound to a load balancing virtual server (Lb_vsr_mysql_master). Three more services (Svc_mysql_2, Svc_mysql_3, and Svc_mysql_4) are created to represent the three slave databases, and they are bound to another load balancing virtual server (Lb_vsr_mysql_slave).

A content switching virtual server (Cs_vsr_mysql_1) is configured with associated policies to send all WRITE requests to the load balancing virtual server, Lb_vsr_mysql_master, and all READ requests to the load balancing virtual server, Lb_vsr_mysql_slave.

When a request reaches the content switching virtual server, the virtual server applies the associated content switching policies to that request. After evaluating the policies, the content switching virtual server routes the request to the appropriate load balancing virtual server, which sends it to the appropriate service.

The following table lists the names and values of the entities and the policy configured on the NetScaler.

Table 1. ***Entity and Policy Names and Values***

Entity Type	Name	IP Address	Protocol	Port	Expression

Entity Type	Name	IP Address	Protocol	Port	Expression
Services	Svc_mysql_1	10.102.29.5	MYSQL	3306	NA
	Svc_mysql_2	10.102.29.6	MYSQL	3306	NA
	Svc_mysql_3	10.102.29.7	MYSQL	3306	NA
	Svc_mysql_4	10.102.29.8	MYSQL	3306	NA
Load balancing virtual servers	Lb_vsr_mysql_master	10.102.29.201	MYSQL	3306	NA
	Lb_vsr_mysql_slave	10.102.29.202	MYSQL	3306	NA
Content switching virtual server	Cs_vsr_mysql_1	10.102.29.161	MYSQL	3306	NA
Content switching policy	Cs_select	NA	NA	NA	"MYSQL.REQ.QUERY.COMMAND.contains(\"select\")"

To configure DataStream for a master/slave database setup by using the command line interface

At the command prompt, type

- add service Svc_mysql_1 10.102.29.5 mysql 3306
- add service Svc_mysql_2 10.102.29.6 mysql 3306
- add service Svc_mysql_3 10.102.29.7 mysql 3306
- add service Svc_mysql_4 10.102.29.8 mysql 3306
- add lb vserver Lb_vsr_mysql_master mysql 10.102.29.201 3306
- add lb vserver Lb_vsr_mysql_slave mysql 10.102.29.202 3306
- bind lb vserver Lb_vsr_mysql_master svc_mysql_1
- bind lb vserver Lb_vsr_mysql_slave svc_mysql_2
- bind lb vserver Lb_vsr_mysql_slave svc_mysql_3
- bind lb vserver Lb_vsr_mysql_slave svc_mysql_4
- add cs vserver Cs_vsr_mysql_1 mysql 10.102.29.161 3306
- add cs policy Cs_select –rule "MYSQL.REQ.QUERY.COMMAND.contains(\"select\")"
- bind cs vserver Cs_vsr_mysql_1 Lb_vsr_mysql_master
- bind cs vserver Cs_vsr_mysql_1 Lb_vsr_mysql_slave –policy Cs_select –priority 10

To configure DataStream for a master/slave database setup by using the configuration utility

Add four services, one to represent the master database server and three to represent the slave database servers.

To add a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.

3. In the Create Service dialog box, specify values for the following parameters.
   - Service Name
   - IP Address
   - Protocol
   - Port
4. Click Create, and then click Close. The service you created appears in the Services pane.

Add two load balancing virtual servers.

To create a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters.
   - Name
   - IP Address
   - Protocol
   - Port
4. Click Create, and then click Close.

Bind the service Svc_mysql_1 to the load balancing virtual server Lb_vsr_mysql_master, and bind the three services (Svc_mysql_2, Svc_mysql_3, and Svc_mysql_4) to the load balancing virtual server Lb_vsr_mysql_slave.

To bind a service to a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind the service (for example, Lb_vsr_mysql_master).
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server (for example, Svc_mysql_1).
5. Click OK.

Create a content switching virtual server.

To add a content switching virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Content Switching) dialog box, specify values for the following parameters.-
   - Name
   - IP Address
   - Protocol
   - Port
4. Click Create, and then click Close.

Create a content switching policy to evaluate all READ requests.

To create a content switching policy by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, click Add.
3. In the Create Content Switching Policy dialog box, in the Name text box, type the name of the policy (for example, Cs_select).
4. Choose the type of policy that you want to create, and configure the policy. To create a rule-based policy, click **Configure**, and do the following:

- In the Create Expression dialog box, choose the expression syntax you want to use and enter your policy expressions.

5. Click Create, and then click Close.

Bind the content switching policy to the content switching virtual server. You should also select a load balancing virtual server as the target for the policy so that, after the content switching virtual server evaluates the policy, it routes requests that match the policy to the load balancing virtual server to forward them to the appropriate database server.

To bind the policy to the content switching virtual server and select a load balancing virtual server target by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, double-click the virtual server to which you want to bind the policy (for example, Cs_vsr_mysql_1).
3. In the Configure Virtual Server (Content Switching) dialog box, on the Policies tab, click Insert Policy, and in the Policy Name column, select the policy that you want to bind to the virtual server (for example, Cs_select).
4. In the Target column next to the policy, select the load balancing virtual server that you want to assign as the target for the policy (for example, Lb_vsr_mysql_slave).
5. Click OK.

Set the load balancing virtual server, Lb_vsr_mysql_master, as the default virtual server for the content switching virtual server by binding the content switching virtual server to this load balancing virtual server. This ensures that the content switching virtual server routes requests that do not match the Cs_select policy to the load balancing virtual server to forward them to the appropriate database server.

To bind the content switching virtual server to a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Content Switching > Policies.
2. In the details pane, double-click the virtual server to which you want to bind the policy (for example, Cs_vsr_mysql_1).
3. In the Configure Virtual Server (Content Switching) dialog box, on the Policies tab, click Insert Policy, and in the Policy Name column, select (Default).
4. In the Target column next to the policy, select the load balancing virtual server that you want to assign as the target for the policy (for example, Lb_vsr_mysql_master).
5. Click OK.

Use Case 2: Configuring the Token Method of Load Balancing for DataStream

May 15, 2015

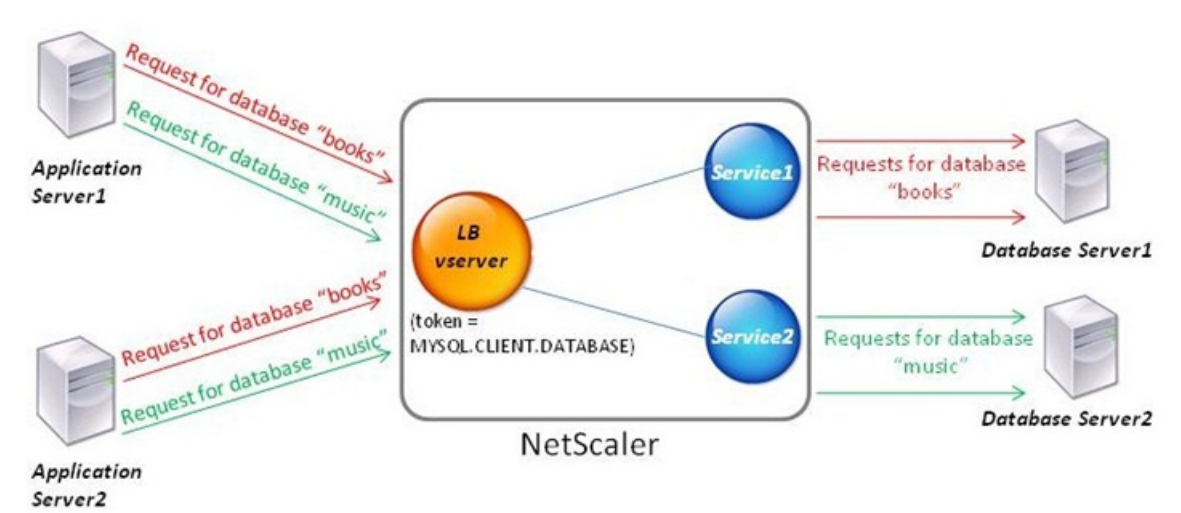
You can configure the token method of load balancing for DataStream to base the selection of database servers on the value of the token extracted from the client (application or web server) requests. These tokens are defined by using SQL expressions. For subsequent requests with the same token, the NetScaler sends the requests to the same database server that handled the initial request. Requests with the same token are sent to the same database server until the maximum connection limit is reached or the session entry has aged out.

You can use the following sample SQL expressions to define tokens:

MySQL	**MS SQL**
MYSQL.REQ.QUERY.TEXT	MSSQL.REQ.QUERY.TEXT
MYSQL.REQ.QUERY.TEXT(n)	MSSQL. REQ.QUERY.TEXT(n)
MYSQL.REQ.QUERY.COMMAND	MSSQL.REQ.QUERY.COMMAND
MYSQL.CLIENT.USER	MSSQL.CLIENT.USER
MYSQL.CLIENT.DATABASE	MSSQL.CLIENT.DATABASE
MYSQL.CLIENT.CAPABILITIES	

The following example shows how the NetScaler DataStream feature works when you configure the token method of load balancing.

Figure 1. How DataStream Works with the Token Method of Load Balancing



In this example, the token is the name of the database. A request with token **books** is sent to **Database Server1** and a request with token **music** is sent to **Database Server2**. All subsequent requests with token **books** are sent to **Database Server1** and requests with token **music** are sent to **Database Server2**. This configuration provides pseudo persistence with the database servers.

To configure this example by using the command line interface

At the command prompt, type:

- add service Service1 192.0.2.9 MYSQL 3306
- add service Service2 192.0.2.11 MYSQL 3306
- add lb vserver token_lb_vserver MYSQL 192.0.2.15 3306 -lbmethod token -rule MYSQL.CLIENT.DATABASE
- bind lb vserver token_lb_vserver Service1
- bind lb vserver token_lb_vserver Service2

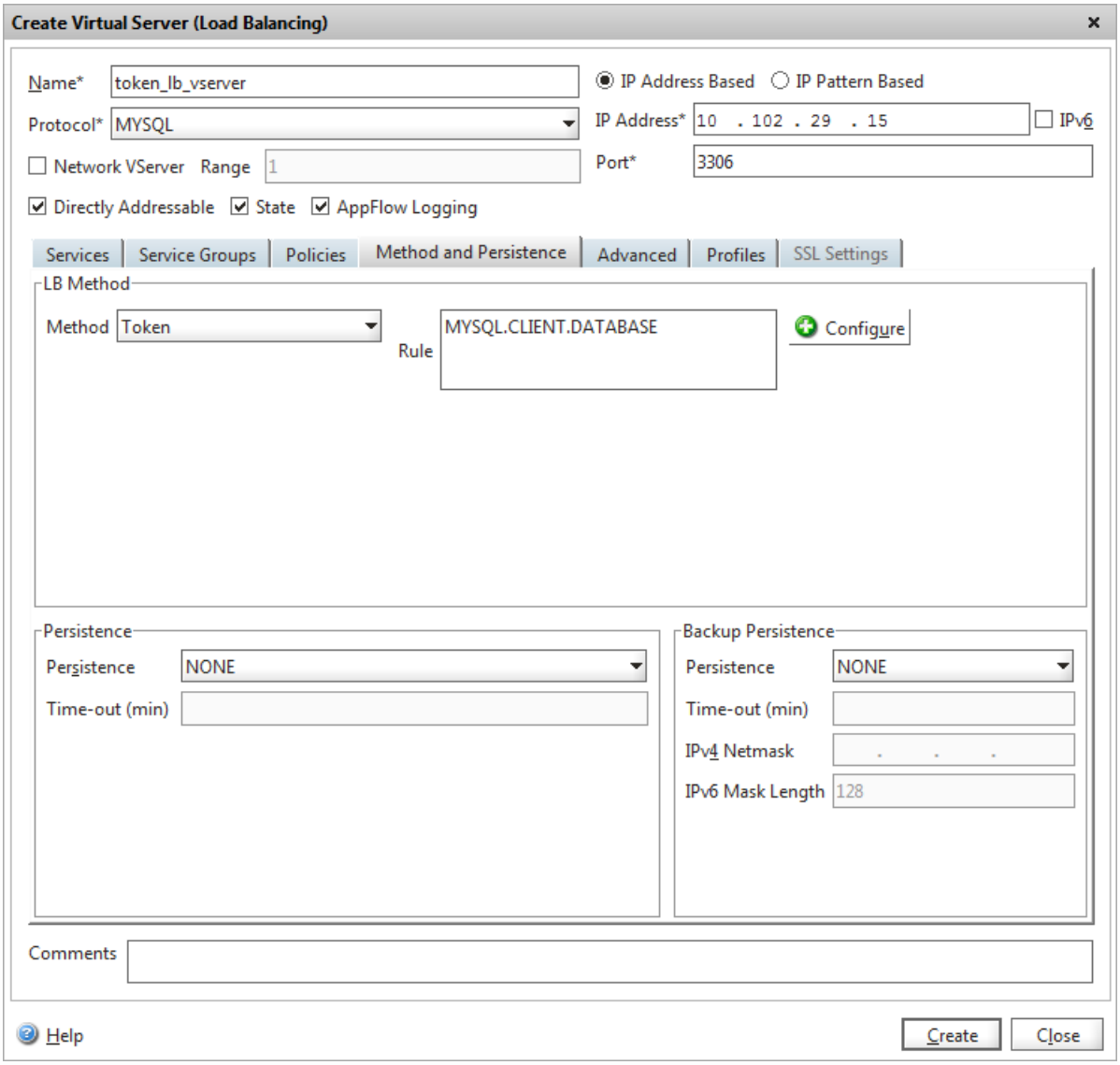
To configure this example by using the configuration utility

1. Add two services to represent the two database servers.
   1. Navigate to Traffic Management > Load Balancing > Services.
   1. In the navigation pane, expand Load Balancing, and then click Services.
   2. In the details pane, click Add.
   3. In the Create Service dialog box, set the following parameters.
      - Service Name
      - IP Address
      - Protocol
      - Port
   4. Click Create, and then click Close.
2. Add one load balancing virtual server and set the token load balancing method.
   1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
   1. In the details pane, click Add.
   2. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters.
      - Name
      - IP Address
      - Protocol

- Port

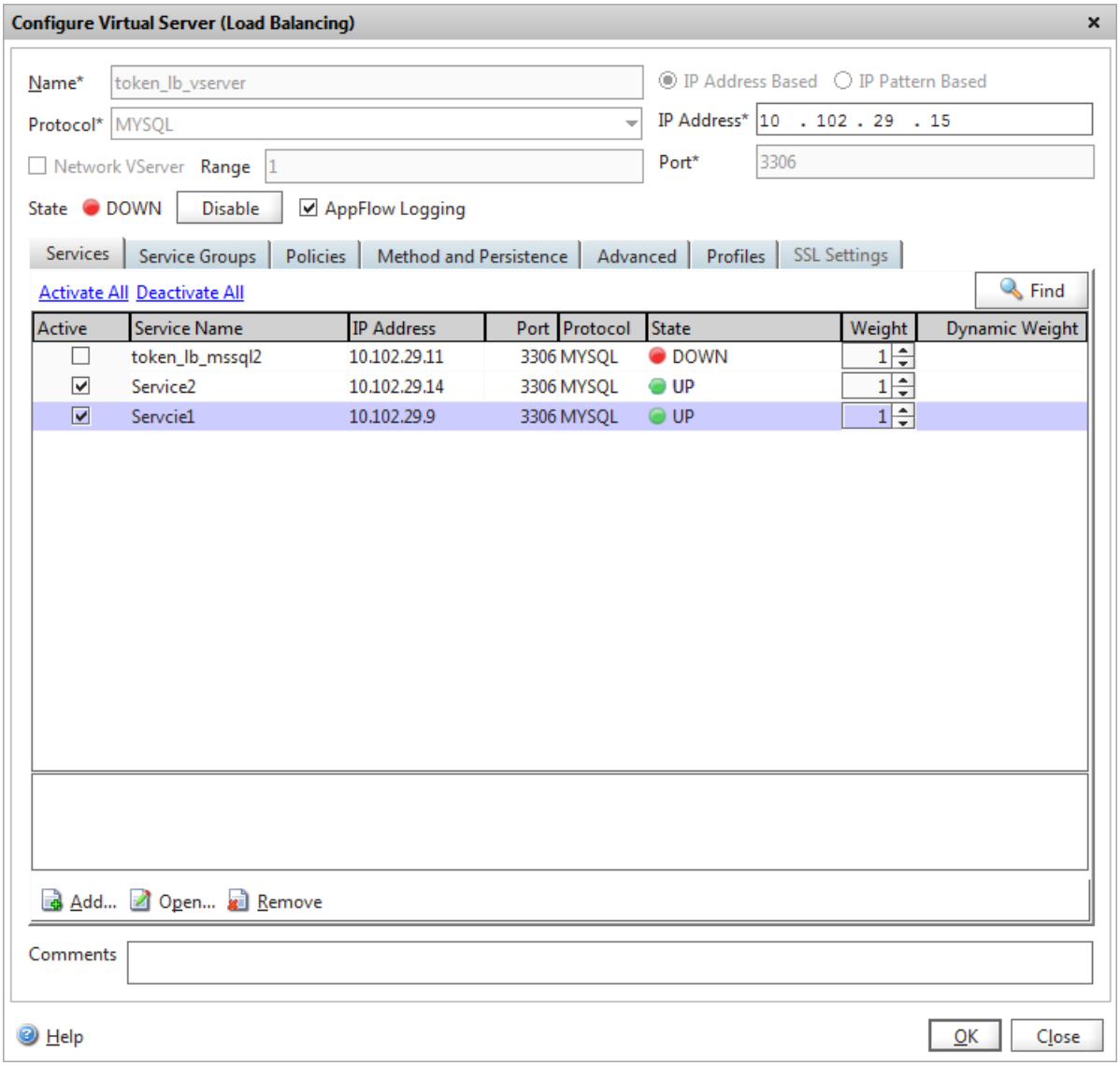
3. On the Method and Persistence tab, under LB Method, in the Method drop list, select Token.
4. In the Rule box, type MYSQL.CLIENT.DATABASE.
5. Click Create, and then click Close.

Figure 2. Configuring a Load Balancing Virtual Server by Using the Configuration Utility



3. Bind the two services to the load balancing virtual server.
   1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
   1. In the details pane, select the virtual server to which you want to bind the service (for example, token_lb_vserver).
   2. Click Open.
   3. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server (for example, Service1).
   4. Click OK.

   Figure 3. Binding Services to a Virtual Server by Using the Configuration Utility



Use Case 3: Logging MSSQL Transactions in Transparent Mode

Jun 09, 2015

You can configure the NetScaler appliance to operate transparently between MSSQL clients and servers, and to only log or analyze details of all client-server transactions. Transparent mode is designed so that the NetScaler appliance only forwards MSSQL requests to the server, and then relays the server's responses to the clients. As the requests and responses pass through the appliance, the appliance logs information gathered from them, as specified by the audit logging or AppFlow configuration, or collects statistics, as specified by the Action Analytics configuration. You do not have to add database users to the appliance.

When operating in transparent mode, the NetScaler appliance does not perform load balancing, content switching, or connection multiplexing for the requests. However, it responds to a client's pre-login packet on behalf of the server so that it can prevent encryption from being agreed upon during the pre-login handshake. The login packet and subsequent packets are forwarded to the server.

This section includes the following details:

- Summary of Configuration Tasks
- Configuring Transparent Mode by Using a Wildcard Virtual Server
- Configuring Transparent Mode by Using MSSQL Services

Summary of Configuration Tasks

Updated: 2015-05-25

For logging or analyzing MSSQL requests in transparent mode, you have to do the following:

- Configure the NetScaler appliance as the default gateway for both clients and servers.
- Do one of the following on the NetScaler appliance:
  - **If you can configure the use source IP address (USIP) option globally,** create a load balancing virtual server with a wildcard IP address and the port number on which the MSSQL servers listen for requests (a port-specific wildcard virtual server). Then, enable the USIP option globally. If you configure a port-specific wildcard virtual server, you do not have to create MSSQL services on the appliance. The appliance discovers the services on the basis of the destination IP address in the client requests. For instructions, see Configuring Transparent Mode by Using a Wildcard Virtual Server.
  - **If you do not want to configure the USIP option globally,** create MSSQL services with the USIP option enabled on each of them. If you configure services, you do not have to create a port-specific wildcard virtual server. For instructions, see Configuring Transparent Mode by Using MSSQL Services.
- Configure audit logging, AppFlow, or Action Analytics to log or collect statistics about the requests. If you configure a virtual server, you can bind your policies either to the virtual server or to the global bind point. If you do not configure a virtual server, you can bind your policies to only the global bind point.

Configuring Transparent Mode by Using a Wildcard Virtual Server

Updated: 2013-11-04

You can configure transparent mode by configuring a port-specific wildcard virtual server and enabling Use Source IP (USIP) mode globally. When a client sends its default gateway (the NetScaler appliance) a request with the IP address of an MSSQL server in the destination IP address header, the appliance checks whether the destination IP address is available. If

the IP address is available, the virtual server forwards the request to the server. Otherwise, it drops the request.

To create a wildcard virtual server by using the command line

At the command prompt, type the following commands to create a wildcard virtual server and verify the configuration:

1. add lb vserver <name> <serviceType> <IPAddress> <port>
2. show lb vserver <name>

Example

```
> add lb vserver wildcardLbVs MSSQL * 1433
 Done
> show lb vserver wildcardLbVs
 wildcardLbVs (*:1433) - MSSQL Type: ADDRESS
 State: UP
 . . .

 Done
>
```

To create a wildcard virtual server by using the NetScaler configuration utility

1. In the navigation pane, expand Traffic Management, expand Load Balancing, and then click Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters:
   - Name*
   - Protocol*
   - IP Address*
   - Port*
4. Click OK.

To enable Use Source IP (USIP) mode globally by using the command line

At the command prompt, type the following commands to enable USIP mode globally and verify the configuration:

- enable ns mode USIP
- show ns mode

Example

```
> enable ns mode USIP
 Done
> show ns mode

 Mode                    Acronym                 Status
 -------                 -------                 ------
          . . .
 3) Use Source IP            USIP                    ON
          . . .
 Done
>
```


To enable USIP mode globally by using the NetScaler configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. Under Modes and Features, click Configure modes.
3. In the Configure Modes dialog box, select the Use Source IP check box.
4. Click OK.

Configuring Transparent Mode by Using MSSQL Services

Updated: 2013-08-23

You can configure transparent mode by configuring MSSQL services and enabling USIP on each service. When a client sends its default gateway (the NetScaler appliance) a request with the IP address of an MSSQL server in the destination IP address header, the appliance forwards the request to the destination server.

To create an MSSQL service and enable USIP mode on the service by using the command line interface

At the command prompt, type the following commands to create an MSSQL service, with USIP enabled, and verify the configuration:

- add service <name> (<IP> | <serverName>) <serviceType> <port> -usip YES
- show service <name>

Example

```
> add service myDBservice 192.0.2.0 MSSQL 1433 -usip YES
 Done
> show service myDBservice
 myDBservice (192.0.2.0:1433) - MSSQL
 State: UP
          . . .
 Use Source IP: YES  Use Proxy Port: YES
          . . .
 Done
>
```

To create an MSSQL service, with USIP enabled, by using the NetScaler configuration utility

1. In the navigation pane, expand Traffic Management, expand Load Balancing, and then click Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters:
   - Service Name*
   - Server*
   - Protocol*
   - Port*

   * A required parameter
4. On the Advanced tab, in the Settings area, select the Override Global check box, and then select the Use Source IP check box.

Use Case 4: Database Specific Load Balancing

Jun 09, 2015

A database server farm should be load balanced not only on the basis of the states of the servers, but also on the basis of the availability of the database on each server. A service might be up, and a load balancing device might show it as being in the UP state, but the requested database might be unavailable on that service. If a query is forwarded to a service on which the database is unavailable, the request is not served. Therefore, a load balancing device must be aware of the availability of a database on each service and, when making a load balancing decision, it must consider only those services on which the database is available.

As an example, consider that database servers server1, server2, and server3 host databases mydatabase1 and mydatabase2. If mydatabase1 becomes unavailable on server2, the load balancing device must be aware of that change in state, and it must load balance requests for mydatabase1 across only server1 and server3. After mydatabase1 becomes available on server2, the load balancing device must include server2 in load balancing decisions. Similarly, if mydatabase2 becomes unavailable on server3, the device must load balance requests for mydatabase2 across only server1 and server2, and it must include server3 in its load balancing decisions only when mydatabase2 becomes available. This load balancing behavior must be consistent across all the databases that are hosted on the server farm.

The Citrix NetScaler appliance implements this behavior by retrieving a list of all the databases that are active on a service. To retrieve the list of active databases, the appliance uses a monitor that is configured with an appropriate SQL query. If the requested database is unavailable on a service, the appliance excludes the service from load balancing decisions until it becomes available. This behavior ensures uninterrupted service to clients.

Note: Database specific load balancing is currently supported for only MSSQL and MySQL service types. This support is also available for Microsoft SQL Server 2012 high availability deployment.
To set up database specific load balancing, you must enable the load balancing feature, configure a load balancing virtual server of type MSSQL or MySQL, configure the services that host the database, and bind the services to the virtual server. The monitor needs valid user credentials to log on to the database server, so you must configure a database user account on each of the servers and then add the user account to the NetScaler appliance. Then, you configure an MSSQL-ECV or MYSQL-ECV monitor and bind the monitor to each service. Finally, you must test the configuration to ensure that it is working as intended. Before you perform these configuration tasks, make sure you understand how database specific load balancing works.

This section includes the following details:

- How Database Specific Load Balancing Works
- Enabling Load Balancing
- Configuring a Load Balancing Virtual Server for Database Specific Load Balancing
- Configuring Services
- Configuring Database Users
- Configuring a Monitor to Retrieve the Names of Active Databases
- HA Group Deployment Support for MSSQL

How Database Specific Load Balancing Works

For database specific load balancing, you configure a monitor that periodically queries each database server for the names of all the active databases on it. The Citrix NetScaler appliance stores the results, and regularly updates the records on the basis of the information retrieved through monitoring. When a client queries a particular database, the appliance uses the configured load balancing method to select a service, and then checks its records to determine whether the database is

available on that service. If the records indicate that the database is not available, it uses the configured load balancing method to select the next available service, and then repeats the check. The appliance forwards the query to the first available service on which the database is active.

Enabling Load Balancing

Updated: 2013-08-08

You can configure load balancing entities such as services and virtual servers when the load balancing feature is disabled, but they will not function until you enable the feature.

To enable load balancing by using the command line interface

At the command prompt, type the following command to enable load balancing and verify the configuration:

- enable ns feature LB
- show ns feature

Example

```
> enable ns feature LoadBalancing
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection         SP              ON
 3)    Load Balancing           LB              ON
 .
 .
 .
 24)   NetScaler Push           push            OFF
 Done
```

To enable load balancing by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Load Balancing check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes.

Configuring a Load Balancing Virtual Server for Database Specific Load Balancing

Updated: 2014-05-30

To configure a virtual server to load balance databases on the basis of availability, you enable the database specific load balancing parameter on the virtual server. Enabling the parameter modifies the load balancing logic so that the NetScaler appliance refers the results of the monitoring probe sent to the selected service, before forwarding the query to that service.

To configure a load balancing virtual server for database specific load balancing

At the command prompt, type the following command to configure a load balancing virtual server for database specific load balancing and verify the configuration:

- add lb vserver <name> <serviceType> <ipAddress> <port> -dbsLb ENABLED
- show lb vserver <name>

Example

In the following example, we create a Microsoft SQL virtual server.

```
> add lb vserver DBSpecificLB1 MSSQL 192.0.2.10 1433 -dbsLb ENABLED
 Done
> show lb vserver DBSpecificLB1
 DBSpecificLB1 (192.0.2.10:1433) - MSSQL Type: ADDRESS
          . . .
 DBS_LB: ENABLED
 Done
>
```

Configuring Services

Updated: 2014-06-03

After you enable the load balancing feature, you must create at least one service for each application server that is to be included in your load balancing setup. The services that you configure provide the connections between the NetScaler appliance and the load balanced servers. Each service has a name and specifies an IP address, a port, and the type of data that is served.

If you create a service without first creating a server object, the IP address of the service is also the name of the server that hosts the service. If you prefer to identify servers by name rather than IP address, you can create server objects and then specify a server's name instead of its IP address when you create a service.

Example

In the following example, we create a service of type MSSQL

```
add service msservice1 5.5.5.5 MSSQL 1433
```

Configuring Database Users

Updated: 2014-10-21

In databases, a connection is always stateful, which means that as soon as a connection is established, it must be authenticated.

You need to configure your database user name and password on the NetScaler ADC. For example, if you have a user John configured on the database, you need to configure the user John on the ADC too. When you add the database user names and passwords on the ADC, these are added to the nsconfig file.
Note: Names are case sensitive.
The ADC uses these user credentials to authenticate the clients, and then authenticate the server connections with the database servers.

To add a database user by using the command line interface

At the command prompt, type

add db user <username> - password <password>
Example

```
> add db user nsdbuser -password dd260427edf
```

To add a database user by using the configuration utility

1. In the details pane, click Add.
2. In the Create Database User dialog box, specify values for the following parameters.
   - User Name
   - Password
   - Confirm Password
3. Click Create, and then click Close. The user you created appears in the Database Users pane.

If you have changed the password of the database user on the database server, you must reset the password of the corresponding user configured on the NetScaler.

To add a database user by using the configuration utility

1. Navigate to System > User Administration > Database Users, and then click Add.

If you have changed the password of the database user on the database server, you must reset the password of the corresponding user configured on the NetScaler.

To reset the password of a database user by using the command line interface

At the command prompt, type

set db user <username> -password <password>
Example

```
> set db user nsdbuser -password dd260538abs
```

To reset the password of database users by using the configuration utility

1. Navigate to System > User Administration > Database Users.
2. In the details pane, select the database user for which you want to reset the password, and then click Open.
3. In the Configure Database User dialog box, modify the values for the following parameters.
   - Password
   - Confirm Password
4. Click OK.

If a database user no longer exists on the database server, you can remove the user from the NetScaler. However, if the user continues to exist on the database server and you remove the user from the NetScaler, any request from the client with this user name does not get authenticated, and therefore, does not get routed to the database server.

To remove a database user by using the command line interface

At the command prompt, type

rm db user <username>
Example

```
>  rm db user nsdbuser
```

To remove a database user by using the configuration utility

1. Navigate to System > User Administration > Database Users.
2. In the details pane, select the database user that you want to remove, and then click Open.
3. In the Proceed message box, click Yes.

Configuring a Monitor to Retrieve the Names of Active Databases

Updated: 2014-05-30

To retrieve a list of all the active databases on a database instance, you create a monitor that logs on to the database server by using a valid user credentials and runs an appropriate SQL query. The SQL query you need to use depends on your SQL server deployment. For example, in a database mirroring setup, you can use the following query to retrieve a list of active databases available on a server instance.

```
select name from sys.databases where state=0
```
You also configure the monitor to evaluate the response for an error condition, and to store the results if there is no error. If the response contains an error, the monitor marks the service as DOWN, and the appliance excludes the service from load balancing decisions until an error is no longer returned.

Note: The database specific load balancing feature is supported only for the MSSQL and MySQL service types. Therefore, the monitor type must be MSSQL-ECV or MYSQL-ECV.

To configure a monitor to retrieve the names of all the active databases hosted on a service by using the command line

At the command prompt, type the following commands to retrieve the names of all the active databases hosted on a service and verify the configuration:

- add lb monitor <monitorName> <type> -userName <string> -sqlQuery <text> -evalRule <expression> -storedb ENABLED
- show lb monitor <monitorName>

Example

In the following example, we create an MSSQL-ECV type monitor.

```
> add lb monitor mssql-monitor1 MSSQL-ECV -userName user1 -sqlQuery "select name
from sys.databases where state=0" -evalRule "MSSQL.RES.TYPE.NE(ERROR)" -storedb ENABLED
 Done
> show lb monitor mssql-monitor1
1)  Name.....: mssql-monitor1  Type.....: MSSQL-ECV       State....:  ENABLED
...
```



```
Special parameters:
  Database.....:""
  User name.....:"user1"
  Query..:select name from sys.databases where state=0
  EvalRule...:MSSQL.RES.TYPE.NE(ERROR)
  Version...:70
  STORE_DB...:ENABLED
 Done
>
```

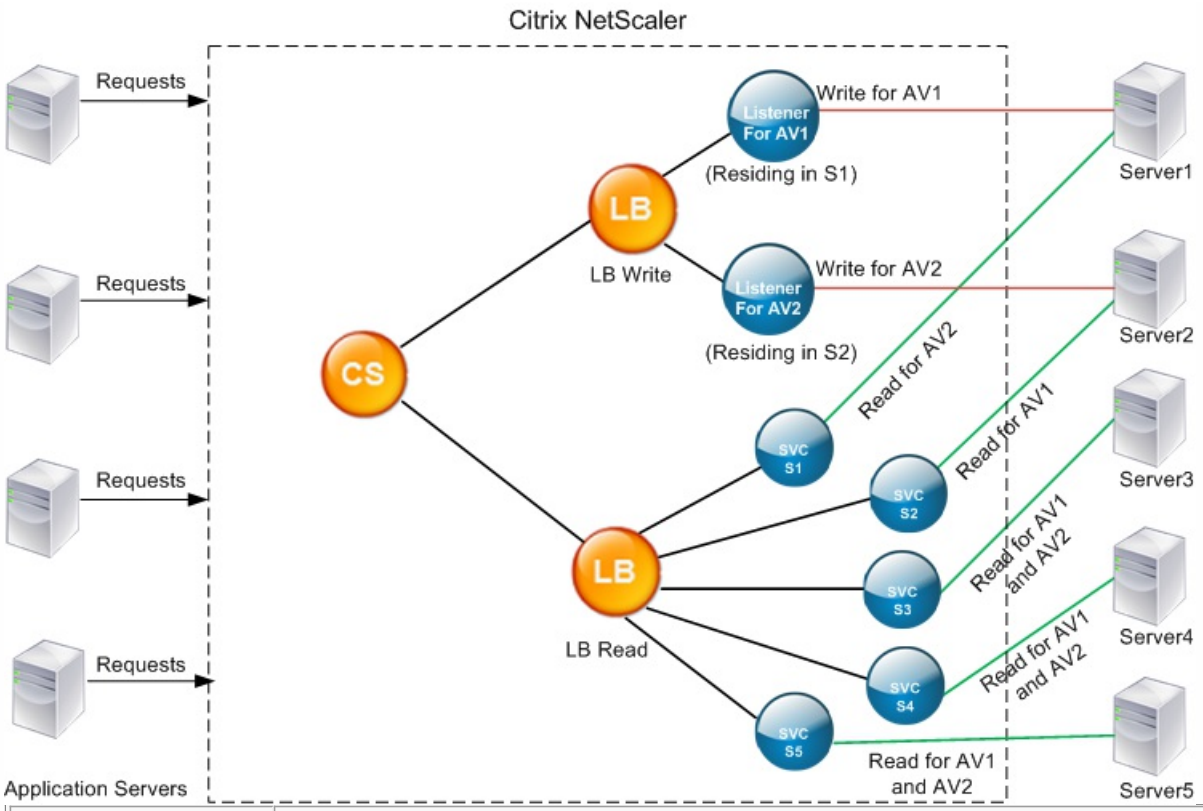
To configure a monitor to retrieve the names of all the active databases hosted on a service by using the configuration utility

1. In the navigation pane, expand Traffic Management, expand Load Balancing, and then click Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, set the following parameters:
   - Name
   - Type
4. On the Special Parameters tab, set the following parameters:
   - Query
   - User Name
   - Rule
   - Store DB
5. Click OK.

HA Group Deployment Support for MSSQL

Updated: 2014-06-17

Consider the following scenario in which database specific load balancing is configured in a high availability group deployment. S1 through S5 are the services on the NetScaler. DB1 through DB4 are the databases on the servers represented by the services S1 through S5. AV1 and AV2 are the availability groups. Each availability group contains up to one primary database server instance and up to four secondary database server instances. A service, representing the servers in the availability group, can be primary for one availability group and secondary for another availability group. Each availability group contains different databases and one listener, which is a service. All requests arrive on the listener service that resides on the primary database. AVI contains databases DB1 and DB2. AV2 contains databases DB3 and DB4. L1 and L2 are the listeners on AV1 and AV2 respectively. S1 is the primary service for AV1 and S2 is the primary service for AV2.



Service	List of Active Databases on the Service
S1	DB1, DB2, DB3, DB4
S2	DB3, DB4
S3	DB3, DB4
S4	DB1, DB2
S5	DB1, DB2

Availability Group	Databases	Services representing the Servers in Availability Group
AV1	DB1, DB2	S1, S4, S5
AV2	DB3, DB4	S1, S2, S3

Queries flow as follows:

1. A READ query for AV1 is load balanced between S4 and S5. S1 is the primary for AV1.
2. A WRITE query for AV1 is directed to L1.
3. A READ query for AV2 is load balanced between S1 and S3. S2 is the primary for AV2.
4. A WRITE query for AV1 is directed to L2.

Sample Configuration

1. Configure load balancing and content switching virtual servers.
   - add lb vserver lbwrite -dbslb enabled
   - add lbvserver lbread MSSQL -dbslb enabled
   - add csvserver csv MSSQL 1.1.1.10 1433
2. Configure two listener services, one for each availability group, and five services S1 through S5 representing databases DB1 through DB4.
   - add service L1 1.1.1.11 MSSQL 1433
   - add service L2 1.1.1.12 MSSQL 1433
   - add service s1 1.1.1.13 MSSQL 1433
   - add service s2 1.1.1.14 MSSQL 1433
   - add service s3 1.1.1.15 MSSQL 1433
   - add service s4 1.1.1.16 MSSQL 1433
   - add service s5 1.1.1.17 MSSQL 1433
3. Bind the services to the load balancing virtual servers.
   - bind lbvserver lbwrite L1
   - bind lbvserver lbwrite L2
   - bind lbvserver lbread s1
   - bind lbvserver lbread s2
   - bind lbvserver lbread s3
   - bind lbvserver lbread s4
   - bind lbvserver lbread s5
4. Configure database users.
   - add db user nsdbuser1 -password dd260427edf
   - add db user nsdbuser2 -password ccd1234xyzw
5. Configure two monitors, monitor_L1 and monitor_L2 for each listener service, to retrieve the list of active databases in that availability group. Add a monitor, monitor1 to retrieve the list of databases for the secondary database server instance.
   - add lb monitor monitor_L1 MSSQL-ECV -userName user1 -sqlQuery "SELECT name FROM sys.databases a INNER JOIN sys.dm_hadr_availability_replica_states b ON a.replica_id=b.replica_id INNER JOIN sys.availability_group_listeners c on b.group_id = c.group_id INNER JOIN sys.availability_group_listener_ip_addresses d on c.listener_id = d.listener_id WHERE b.role = 1 and d.ip_address like '1.1.1.11'" -evalRule "MSSQL.RES.TYPE.NE(ERROR)" –storedb ENABLED
   - add lb monitor monitor_L2 MSSQL-ECV -userNameuser1 -sqlQuery "SELECT name FROM sys.databases a INNER JOIN sys.dm_hadr_availability_replicca_states b ON a.replica_id=b.replica_id INNER JOIN sys.availability_group_listeners c on b.group_id = c.group_id INNER JOIN sys.availability_group_listener_ip_addresses d on c.listener_id = d.listener_id WHERE b.role = 1 and d.ip_address like '1.1.1.12'" -evalRule "MSSQL.RES.TYPE.NE(ERROR)" -storedb ENABLED
   - add lb monitor monitor1 MSSQL-ECV -userNameuser1 -sqlQuery "SELECT name FROM sys.databases a INNER JOIN sys.dm_hadr_availability_replica_states b ON a.replica_id=b.replica_id WHERE b.role = 2" -evalRule "MSSQL.RES.TYPE.NE(ERROR)" -storedb ENABLED
6. Configure read and write policies.
   - add cs policy pol_write -rule "MSSQL.REQ.QUERY.TEXT.CONTAINS(\"insert\")"
   - add cs policy pol_read -rule "MSSQL.REQ.QUERY.TEXT.CONTAINS(\"select\")"
7. Bind the policies to the content switching virtual server.
   - bind csvserver csv -targetLBVserver lbwrite -policyName pol_write -priority 11
   - bind csvserver csv -targetLBVserver lbread -policyName pol_read -priority 12

8. Bind monitors to the services. Bind monitors to services L1 and L2 to get the list of active databases for the availability group for which it is the listener. Bind monitors to all the services that are bound to the read-only virtual server.
   - bind service L1 -monitorName monitor_L1
   - bind service L2 -monitorName monitor_L2
   - bind service s1 -monitorName monitor1
   - bind service s2 -monitorName monitor1
   - bind service s3 -monitorName monitor1
   - bind service s4 -monitorName monitor1
   - bind service s5 -monitorName monitor1

DataStream Reference

Mar 30, 2012

This reference describes the MySQL and TDS protocols, the database versions, the authentication methods, and the character sets supported by the DataStream feature. It also describes how NetScaler handles transaction requests and special queries that modify the state of a connection.

You can also configure the NetScaler appliance to generate audit log messages for the DataStream feature.

Supported Database Versions, Protocols, and Authentication Methods

Updated: 2014-08-28

	MySQL Database	**MS SQL Database**
Database Versions	MySQL database versions 4.1, 5.0, 5.1, 5.4, 5.5, and 5.6.	MS SQL database versions 2000, 2000SP1, 2005, 2008, 2008R2, and 2012.
Protocols	MySQL protocol version 10. For information about the MySQL protocol, see http://dev.mysql.com/doc/internals/en/client-server-protocol.html	TDS protocol version 7.1 and higher. For information about the TDS protocol, see http://msdn.microsoft.com/en-us/library/dd304523(v=prot.13).aspx
Authentication Methods	MySQL native authentication is supported.	SQL server authentication and Windows Authentication (Kerberos) are supported.

Character Sets

The DataStream feature supports only the UTF-8 charset.

The character set used by the client while sending a request may be different from the character set used in the database server responses. Although the charset parameter is set during the connection establishment, it can be changed at any time by sending an SQL query. The character set is associated with a connection, and therefore, requests on connections with one character set cannot be multiplexed onto a connection with a different character set.

NetScaler parses the queries sent by the client and the responses sent by the database server.

The character set associated with a connection can be changed after the initial handshake by using the following two queries:

- SET NAMES <charset> COLLATION <collation>
- SET CHARACTER SET <charset>

Transactions

In MySQL, transactions are identified by using the connection parameter AUTOCOMMIT or the BEGIN:COMMIT queries. The AUTOCOMMIT parameter can be set during the initial handshake, or after the connection is established by using the

query SET AUTOCOMMIT.

NetScaler explicitly parses each and every query to determine the beginning and end of a transaction.

In MySQL protocol, the response contains two flags to indicate whether the connection is a transaction, the TRANSACTION and AUTOCOMMIT flags.

If the connection is a transaction, the TRANSACTION flag is set. Or, if the AutoCommit mode is OFF, the AUTOCOMMIT flag is not set. NetScaler parses the response, and if either the TRANSACTION flag is set or the AUTOCOMMIT flag is not set, it does not do connection multiplexing. When these conditions are no longer true, the NetScaler begins connection multiplexing.

Special Queries

Updated: 2014-05-21

There are special queries, such as SET and PREPARE, that modify the state of the connection and may break request switching, and therefore, these need to be handled differently.

On receiving a request with special queries, NetScaler sends an OK response to the client and additionally, stores the request in the connection.

When a non-special query, such as INSERT and SELECT, is received along with a stored query, the NetScaler first, looks for the server-side connection on which the stored query has already been sent to the database server. If no such connections exist, NetScaler creates a new connection, and sends the stored query first, and then, sends the request with the non-special query.

In case of SET, USE db, and INIT_DB special queries, the appliance modifies a field in the server side connection corresponding to the special query. This results in better reuse of the server side connection.

Only 16 queries are stored in each connection.

The following is a list of the special queries for which NetScaler has a modified behavior.

SET query

The SET SQL queries define variables that are associated with the connection. These queries are also used to define global variables, but as of now, NetScaler is unable to differentiate between local and global variables. For this query, the NetScaler uses the 'store and forward' mechanism described earlier .

USE <db> query

Using this query, the user can change the database associated with a connection. In this case, NetScaler parses the <db> value sent and modifies a field in the server side connection to reflect the new database to be used.

INIT_DB command

Using this query, the user can change the database associated with a connection. In this case, NetScaler parses the <init_db> value sent and modifies a field in the server side connection to reflect the new database to be used.

COM_PREPARE

NetScaler stops request switching on receiving this command.

PREPARE query

This query is used to create prepared statements that are associated with a connection. For this query, the NetScaler uses the 'store and forward' mechanism described earlier in this section.

Audit Log Message Support

Updated: 2013-09-30

You can now configure the NetScaler appliance to generate audit log messages for the DataStream feature. Audit log messages are generated when client-side and server-side connections are established, closed, or dropped. The categories of messages that you can log and view are ERROR and INFO. Error messages for client-side connections begin with "CS" and error messages for server-side connections begin with "SS." Additional information is provided where necessary. For example, log messages for closed connections (CS_CONN_CLOSED) include only the connection ID. However, log messages for established connections (CS_CONN_ESTD) include information such as the user name, database name, and the client IP address in addition to the connection ID.

Domain Name System

May 26, 2015

You can configure the Citrix NetScaler appliance to function as an authoritative domain name server (ADNS server) for a domain. You can add the DNS resource records that belong to the domain for which the appliance is authoritative and configure resource record parameters. You can also configure the NetScaler appliance as a proxy DNS server that load balances a farm of DNS name servers that are either within your network or outside your network. You can configure the appliance as an end resolver and forwarder. You can configure DNS suffixes that enable name resolution when fully qualified domain names are not configured. The appliance also supports the DNS ANY query that retrieves all the records that belong to a domain.

You can configure the NetScaler appliance to concurrently function as an authoritative DNS server for one domain and a DNS proxy server for another domain. When you configure the NetScaler as the authoritative DNS server or DNS proxy server for a zone, you can enable the appliance to use the Transmission Control Protocol (TCP) for response sizes that exceed the size limit specified for the User Datagram Protocol (UDP).

How DNS Works on the NetScaler

You can configure the NetScaler appliance to function as an ADNS server, DNS proxy server, end resolver, and forwarder. You can add DNS resource records on the NetScaler, including service (SRV) records, IPv6 (AAAA) records, address (A) records, mail exchange (MX) records, canonical name (CNAME) records, pointer (PTR) records, start of authority (SOA) records, and text (TXT) records. Also, you can configure the NetScaler to load balance external DNS name servers.

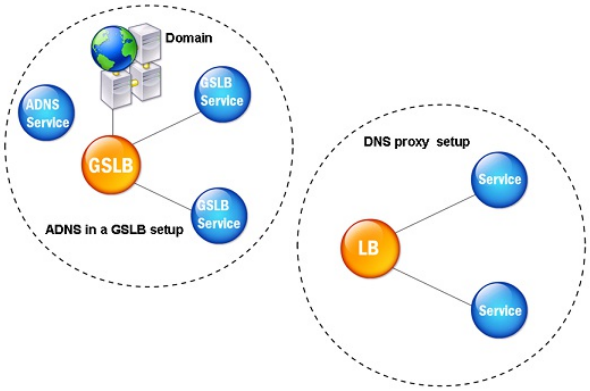
The NetScaler can be configured as the authority for a domain. To do this, you add valid SOA and NS records for the domain.

An ADNS server is a DNS server that contains complete information about a zone.

To configure the NetScaler as an ADNS server for a zone, you must add an ADNS service, and then configure the zone. To do so, you add valid SOA and NS records for the domain. When a client sends a DNS request, the NetScaler appliance searches the configured resource records for the domain name. You can configure the ADNS service to be used with the NetScaler Global Server Load Balancing (GSLB) feature.

You can delegate a subdomain, by adding NS records for the subdomain to the zone of the parent domain. You can then make the NetScaler authoritative for the subdomain, by adding a "glue record" for each of the subdomain name servers. If GSLB is configured, the NetScaler makes a GSLB load balancing decision based on its configuration and replies with the IP address of the selected virtual server. The following figure shows the entities in an ADNS GSLB setup and a DNS proxy setup.

Figure 1. DNS Proxy Entity Model



The NetScaler appliance can function as a DNS proxy. Caching of DNS records, which is an important function of a DNS proxy, is enabled by default on the NetScaler appliance. This enables the NetScaler to provide quick responses for repeated translations. You must also create a load balancing DNS virtual server, and DNS services, and then bind these services to the virtual server.

The NetScaler provides two options, minimum time to live (TTL) and maximum TTL for configuring the lifetime of the cached data. The cached data times out as specified by your settings for these two options. The NetScaler checks the TTL of the DNS record coming from the server. If the TTL is less than the configured minimum TTL, it is replaced with the configured minimum TTL. If the TTL is greater than the configured maximum TTL, it is replaced with the configured maximum TTL.

The NetScaler also allows caching of negative responses for a domain. A negative response indicates that information about a requested domain does not exist, or that the server cannot provide an answer for the query. The storage of this information is called *negative caching*. Negative caching helps speed up responses to queries on a domain, and can optionally provide the record type.

A negative response can be one of the following:

- NXDOMAIN error message - If a negative response is present in the local cache, the NetScaler returns an error message (NXDOMAIN). If the response is not in the local cache, the query is forwarded to the server, and the server returns an NXDOMAIN error to the NetScaler. The NetScaler caches the response locally, then returns the error message to the client.
- NODATA error message - The NetScaler sends a NODATA error message, if the domain name in query is valid but records of the given type are not available.

The NetScaler supports recursive resolution of DNS requests. In recursive resolution, the resolver (DNS client) sends a recursive query to a name server for a domain name. If the queried name server is authoritative for the domain, it responds with the requested domain name. Otherwise, the NetScaler queries the name servers recursively until the requested domain name is found.

Before you can apply the recursive query option, you must first enable it. You can also set the number of times the DNS resolver must send a resolution request (DNS retries) if a DNS lookup fails.

You can configure the NetScaler as a DNS forwarder. A forwarder passes DNS requests to external name servers. The NetScaler allows you to add external name servers and provides name resolution for domains outside the network. The NetScaler also allows you to set the name lookup priority to DNS or Windows Internet Name Service (WINS).

Round Robin DNS

When a client sends a DNS request to find the DNS resource record, it receives a list of IP addresses resolving to the name in the DNS request. The client then uses one of the IP addresses in the list, generally, the first record or IP address. Hence, a single server is used for the total TTL of the cache and is overloaded when a large number of requests arrive.

When the NetScaler receives a DNS request, it responds by changing the order of the list of DNS resource records in a round robin method. This feature is called *round robin DNS*. Round robin distributes the traffic equally between data centers. The NetScaler performs this function automatically. You do not have to configure this behavior.

Functional Overview

If the NetScaler is configured as an ADNS server, it returns the DNS records in the order in which the records are configured. If the NetScaler is configured as a DNS proxy, it returns the DNS records in the order in which it receives the records from the server. The order of the records present in the cache matches the order in which records are received from the server.

The NetScaler then changes the order in which records are sent in the DNS response in a round robin method. The first response contains the first record in sequence, the second response contains the second record in sequence, the third response contains the third record in sequence, and the order continues in the same sequence. Thus, clients requesting the same name can connect to different IP addresses.

Round Robin DNS Example

As an example of round robin DNS, consider DNS records that have been added as follows:

```
add dns addRec ns1 1.1.1.1  add dns addRec ns1 1.1.1.2  add dns addRec ns1 1.1.1.3  add dns addRec ns1 1.1.1.4
```

The domain, abc.com is linked to an NS record as follows:

```
add dns nsrec abc.com. ns1
```

When the NetScaler receives a query for the A record of ns1, the Address records are served in a round robin method as follows. In the first DNS response, 1.1.1.1 is served as the first record:

```
ns1.            1H IN A     1.1.1.1  ns1.            1H IN A     1.1.1.2  ns1.            1H IN A     1.1.1.3  ns1.            1H IN A     1.1.1.4
```

In the second DNS response, the second IP address, 1.1.1.2 is served as the first record:

```
ns1.            1H IN A     1.1.1.2  ns1.            1H IN A     1.1.1.3  ns1.            1H IN A     1.1.1.4  ns1.            1H IN A     1.1.1.1
```

In the third DNS response, the third IP address, 1.1.1.2 is served as the first record:

```
ns1.            1H IN A     1.1.1.3  ns1.            1H IN A     1.1.1.4  ns1.            1H IN A     1.1.1.1  ns1.            1H IN A     1.1.1.2
```

Configuring DNS Resource Records

Jun 19, 2014

You configure resource records on the Citrix® NetScaler® appliance when you configure the appliance as an ADNS server for a zone. You can also configure resource records on the appliance if the resource records belong to a zone for which the appliance is a DNS proxy server. On the appliance, you can configure the following record types:

- Service records
- AAAA records
- Address records
- Mail Exchange records
- Name Server records
- Canonical records
- Pointer records
- NAPTR records
- Start of Authority records
- Text records

The following table lists the record types and the number of records (per record type) that you can configure for a domain on the NetScaler.

Table 1. Record Type and Number Configurable

Record Type	Number of Records
Address (A)	25
IPv6 (AAAA)	5
Mail exchange (MX)	12
Name server (NS)	16
Service (SRV)	8
Pointer (PTR)	20
Canonical name (CNAME)	1
Start of Authority (SOA)	1
Text (TXT)	20

Record Type	Number of Records
Naming Authority Pointer (NAPTR)	20

Creating SRV Records for a Service

Nov 21, 2014

The SRV record provides information about the services available on the NetScaler appliance. An SRV record contains the following information: name of the service and the protocol, domain name, TTL, DNS class, priority of the target, weight of records with the same priority, port of the service, and host name of the service. The NetScaler chooses the SRV record that has the lowest priority setting first. If a service has multiple SRV records with the same priority, clients use the weight field to determine which host to use.

To add an SRV record by using the command line interface

At the command prompt, type the following commands to add an SRV record and verify the configuration:

- add dns srvRec <domain> <target> -priority <positive_integer> -weight <positive_integer> -port <positive_integer> [-TTL <secs>]
- sh dns srvRec <domain>

Example

```
> add dns srvRec _http._tcp.example.com nameserver1.com -priority 1 -weight 1 -port 80
 Done
> show dns srvRec _http._tcp.example.com
1)     Domain Name : _http._tcp.example.com
       Target Host : nameserver1.com
       Priority : 1    Weight : 1
       Port : 80       TTL : 3600 secs
 Done
>
```

To modify or remove an SRV record by using the command line interface

- To modify an SRV record, type the set dns srvRec command, the name of the domain for which the SRV record is configured, the name of the target host that hosts the associated service, and the parameters to be changed, with their new values.
- To remove an SRV record, type the rm dns srvRec command, the name of the domain for which the SRV record is configured, and the name of the target host that hosts the associated service.

To configure an SRV record by using the configuration utility

1. Navigate to Traffic Management > DNS > Records > SRV Records.
2. In the details pane, do one of the following:
   - To create a new SRV record, click Add.
   - To modify an existing SRV record, select the SRV record, and then click Open.
3. In the Create SRV Record or Configure SRV Record dialog box, specify values for the following parameters as shown:

   - Domain Name*—domain (cannot be changed for an existing SRV record)
   - Target*—target (cannot be changed for an existing SRV record)
   - Priority*—priority
   - Weight*—weight

- Port*—port
- TTL—TTL

* A required parameter

4. Click Create or OK.

Creating AAAA Records for a Domain Name

Nov 21, 2014

An AAAA resource record stores a single IPv6 address.

To add an AAAA record by using the command line interface

At the command prompt, type the following commands to add an AAAA record and verify the configuration:

- add dns aaaaRec <hostName> <IPv6Address> ... [-TTL <secs>]
- show dns aaaaRec <hostName>

Example

```
> add dns aaaaRec www.example.com 2001:0db8:0000:0000:0000:0000:1428:57ab
 Done
> show dns aaaaRec www.example.com
1)     Host Name : www.example.com
       Record Type : ADNS            TTL : 5 secs
       IPV6 Address : 2001:db8::1428:57ab
 Done
>
```

To remove an AAAA record and all of the IPv6 addresses associated with the domain name, type the rm dns aaaaRec command and the domain name for which the AAAA record is configured. To remove only a subset of the IPv6 addresses associated with the domain name in an AAAA record, type the rm dns aaaaRec command, the domain name for which the AAAA record is configured, and the IPv6 addresses that you want to remove.

To add an AAAA record by using the configuration utility

1. Navigate to Traffic Management > DNS > Records > AAAA Records.
2. In the details pane, click Add.
3. In the Create AAAA Record dialog box, specify values for the following parameters as shown:

   - Host Name*—hostName
   - IPv6 Address*—IPv6Address
   - TTL-TTL

   * A required parameter

4. Click Add. The IPv6 address appears in the IP box.
5. Click Create, and then click Close.

Creating Address Records for a Domain Name

Nov 21, 2014

Address (A) records are DNS records that map a domain name to an IPv4 address.

You cannot delete Address records for a host participating in global server load balancing (GSLB). However, the NetScaler deletes Address records added for GSLB domains when you unbind the domain from a GSLB virtual server. Only user-configured records can be deleted manually. You cannot delete a record for a host referenced by records such as NS, MX, or CNAME.

To add an Address record by using the command line interface

At the command prompt, type the following commands to add an Address record and verify the configuration:

- add dns addRec <hostName> <IPAddress> [-TTL <secs>]
- show dns addRec <hostName>

Example

```
> add dns addRec ns.example.com 192.0.2.0
 Done
> show dns addRec ns.example.com
1)     Host Name : ns.example.com
       Record Type : ADNS                TTL : 5 secs
       IP Address : 192.0.2.0
 Done
>
```

To remove an Address record and all of the IP addresses associated with the domain name, type the rm dns addRec command and the domain name for which the Address record is configured. To remove only a subset of the IP addresses associated with the domain name in an Address record, type the rm dns addRec command, the domain name for which the Address record is configured, and the IP addresses that you want to remove.

To add an Address record by using the configuration utility

1. Navigate to Traffic Management > DNS > Records > Address Records
2. In the details pane, click Add.
3. Click Add. The IP address appears in the IP Address box.
4. Click Create, and then click Close.

Creating MX Records for a Mail Exchange Server

Nov 21, 2014

Mail Exchange (MX) records are used to direct email messages across the Internet. An MX record contains an MX preference that specifies the MX server to be used. The MX preference values range from 0 through 65536. An MX record contains a unique MX preference number. You can set the MX preference and the TTL values for an MX record.

When an email message is sent through the Internet, a mail transfer agent sends a DNS query requesting the MX record for the domain name. This query returns a list of host names of mail exchange servers for the domain, along with a preference number. If there are no MX records, the request is made for the Address record of that domain. A single domain can have multiple mail exchange servers.

To add an MX record by using the command line interface

At the command prompt, type the following commands to add an MX record and verify the configuration:

- add dns mxRec <domain> -mx <string> -pref <positive_integer> [-TTL <secs>]
- show dns mxRec <domain>

Example

```
> add dns mxRec example.com -mx mail.example.com -pref 1
 Done
> show dns mxRec example.com
1)     Domain : example.com    MX Name : mail.example.com
       Preference :   1       TTL : 5 secs
 Done
>
```

To modify or remove an MX record by using the command line interface

- To modify an MX record, type the set dns mxRec command, the name of the domain for which the MX record is configured, the name of the MX record, and the parameters to be changed, with their new values.
- To set the TTL parameter to its default value, type the unset dns mxRec command, the name of the domain for which the MX record is configured, the name of the MX record, and -TTL without any TTL value. You can use the unset dns mxRec command to unset only the TTL parameter.
- To remove an MX record, type the rm dns mxRec command, the name of the domain for which the MX record is configured, and the name of the MX record.

To add an MX record by using the configuration utility

1. Navigate to Traffic Management > DNS > Records > Mail Exchange Records
2. In the details pane, do one of the following:
   - To create a new MX record, click Add.
   - To modify an existing MX record, select the MX record, and then click Open.
3. In the Create Mail Exchange Record or Configure Mail Exchange Record dialog box, specify values for the following parameters as shown:

   - Domain Name*—domain (cannot be changed for an existing MX record)

- Mail Exchange*—mx (cannot be changed for an existing MX record)
- Preference No.*—pref
- TTL—TTL

* A required parameter

4. Click Create or OK.

Creating NS Records for an Authoritative Server

Nov 21, 2014

Name Server (NS) records specify the authoritative server for a domain. You can configure a maximum of 16 NS records. You can use an NS record to delegate the control of a subdomain to a DNS server.

To create an NS record by using the command line interface

At the command prompt, type the following commands to create an NS record and verify the configuration:

- add dns nsRec <domain> <nameServer> [-TTL <secs>]
- show dns nsRec <domain>

Example

```
> add dns nsRec example.com nameserver1.example.com
 Done
> show dns nsRec example.com
1)     Domain : example.com    NameServer : nameserver1.example.com
       TTL : 5 sec
 Done
>
```

To remove an NS record, type the rm dns nsRec command, the name of the domain to which the NS record belongs, and the name of the name server.

To create an NS record by using the configuration utility

1. Navigate to Traffic Management > DNS > Records > Name Server Records
2. In the details pane, click Add.
3. In the Create Name Server Record dialog box, specify values for the following parameters as shown:

   - Domain Name*—domain
   - Name Server*—nameServer
   - TTL—TTL

   * A required parameter

4. Click Create, and then click Close.

Creating CNAME Records for a Subdomain

May 26, 2015

A canonical name record (CNAME record) is an alias for a DNS name. These records are useful when multiple services query the DNS server. The host that has an address (A) record cannot have a CNAME record.

In some cases, a NetScaler appliance in proxy mode requests an address record from the cache instead of the server.

To add a CNAME record by using the command line interface

At the command prompt, type the following commands to create a CNAME record and verify the configuration:

- add dns cnameRec <aliasName> <canonicalName> [-TTL <secs>]
- show dns cnameRec <aliasName>

Example

```
> add dns cnameRec www.example.com www.examplenw.com
 Done
> show dns cnameRec www.example.com
      Alias Name     Canonical Name  TTL
1)    www.example.com       www.examplenw.com      5 secs
 Done
>
```

To remove a CNAME record for a given domain, type the rm dns cnameRec command and the alias of the domain name.

To add a CNAME record by using the configuration utility

1. Navigate to Traffic Management > DNS > Records > Canonical Records
2. In the details pane, click Add.
3. In the Create Canonical Name Record dialog box, specify values for the following parameters as shown:

   - Alias Name*—aliasName
   - Canonical Name*—canonicalName
   - TTL—TTL

   * A required parameter

4. Click Create, and then click Close.

Caching of CNAME Records

Updated: 2015-05-26

NetScaler ADC when deployed in a proxy mode does not always send the query for an address record to the back-end server. This happens when for a answer to a query for an address record, a partial CNAME chain is present in the cache. There are few conditions in which the ADC caches the partial CNAME record and serves the query from the cache. Following are the conditions:

- NetScaler should be deployed in a proxy mode
- The response from the back-end server should have a CNAME chain, for which the record type of last entry in the answer section must be a CNAME and the question type not a CNAME

- The response from the back-end server cannot be a No-data or NX-Domain
- The response from the back-end server has to be a authoritative response

Creating NAPTR Records for Telecommunications Domain

Oct 29, 2014

NAPTR (Naming Address Pointer) is one of the most commonly used DNS record in telecommunications domain. NAPTR records map the Internet telephony address space to the Internet address space. They therefore enable a mobile device to send a request to the correct server. The combination of NAPTR records with Service Records (SRV) allows the chaining of multiple records to form complex rewrite rules that produce new domain labels or uniform resource identifiers (URIs). The DNS code for NAPTR is 35.

NetScaler ADCs support NAPTR in two modes: ADNS mode and proxy mode. In proxy mode, the ADC caches the response from the servers and uses the cached records to server future queries. A maximum of 20 NAPTR records can be added for a particular domain in NetScaler. NetScaler caches the reply to a DNS NAPTR record query. Any subsequent requests for the NAPTR record is served from the cache.

To create a NAPTR record by using command line interface

At the command prompt, type the following commands to add a NAPTR record and verify the configuration:

add dns naptrRec <order> <preference>[flags<string>][services<string>](regexp<expressions>|-replacement<string>) [-TTL<secs>]

To remove a NAPTR record by using command line interface

rm dns naptrRec<domain> (<order> <preference> [-flags <string>] [-services <string>] (-regexp <expression> | -replacement <string>)) | -recordId <positive_integer>@)

To configure a NAPTR record using configuration utility

1. In the navigation pane, expand DNS, expand Records, and then click NAPTR Records.
2. In the Create NAPTR Record dialog box, set the following parameters:
   - Domain—Name of the domain for the NAPTR record. Maximum length: 255 characters.
   - Order—Integer specifying the order in which the NAPTR records must be processed in order to accurately represent the ordered list of rules. The ordering is from lowest to highest. Maximum value: 65535.
   - Preference— Integer specifying the preference of this NAPTR among NAPTR records. Maximum value: 65535.
   - Flags— Flags for this NAPTR record. Maximum length: 255 characters.
   - Regular Expression—Regular expression that specifies the substitution expression for this NAPTR. Maximum length: 255 characters.
   - Replacement—The replacement domain name for this NAPTR record. Maximum length: 255 characters.
   - TTL— Time to Live (TTL), in seconds, for the record.
3. Click Create, and then click Close.

-
-

```
> add dns ptrRec 0.2.0.192.in-addr.arpa example.com
 Done
> show dns ptrRec 0.2.0.192.in-addr.arpa
1)     Reverse Domain Name : 0.2.0.192.in-addr.arpa
       Domain Name : example.com            TTL : 3600 secs
 Done
>
```

-
-
-
-

-
-

```
> add dns soaRec example.com -originServer nameserver1.example.com -contact admin.example.com
 Done
> show dns soaRec example.com
1)     Domain Name : example.com
       Origin Server : nameserver1.example.com
       Contact : admin.example.com
       Serial No. : 100       Refresh : 3600 secs     Retry : 3 secs
       Expire : 3600 secs      Minimum : 5 secs       TTL : 3600 secs
 Done
>
```

-
-

  -
  -

  -
  -
  -

-
-

```
> add dns txtRec www.example.com "Contact: Mark" "Email: mark@example.com" -TTL 36000
 Done
> show dns txtRec www.example.com
1) Domain : www.example.com Record id: 13783    TTL : 36000 secs Record Type : ADNS
  "Contact: Mark"
  "Email: mark@example.com"
 Done
```

-
-

```
> show dns txtRec www.example.com
1) Domain : www.example.com Record id: 36865    TTL : 36000 secs Record Type : ADNS
  "Contact: Evan"
  "Email: evan@example.com"
2) Domain : www.example.com Record id: 14373    TTL : 36000 secs Record Type : ADNS
```

"Contact: Mark"
"Email: mark1@example.com"
Done

>rm dns txtRec www.example.com -recordID 36865
Done
> show dns txtRec www.example.com
1) Domain : www.example.com Record id: 14373 TTL : 36000 secs Record Type : ADNS
"Contact: Mark"
"Email: mark1@example.com"
Done

-
-

-

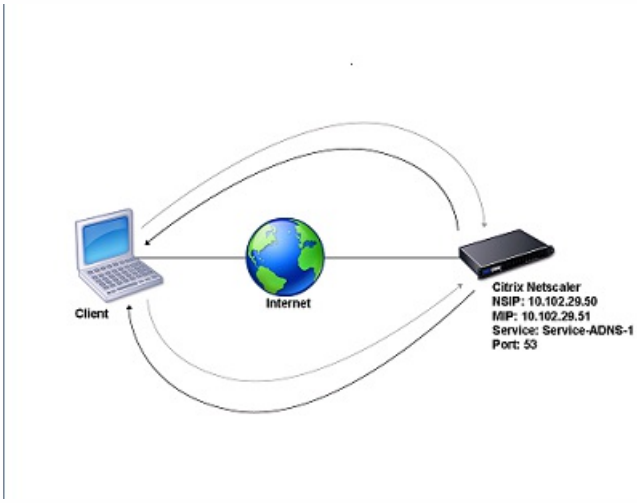
```
> stat dns
DNS Statistics

Runtime Statistics
Dns queries                    21
NS queries                     8
SOA queries                    18
.
.
.
Configuration Statistics
AAAA records                   17
A records                      36
MX records                     9
.
.
.
Error Statistics
Nonexistent domain             17
No AAAA records                0
No A records                   13
.
.
.
Done
>
```

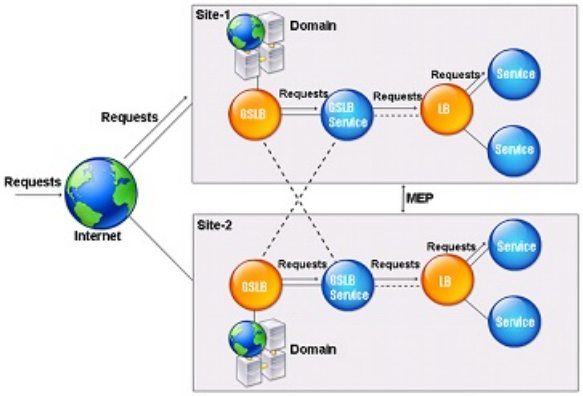
-
-
-
-
-

```
> add dns zone example.com -proxyMode Yes
```

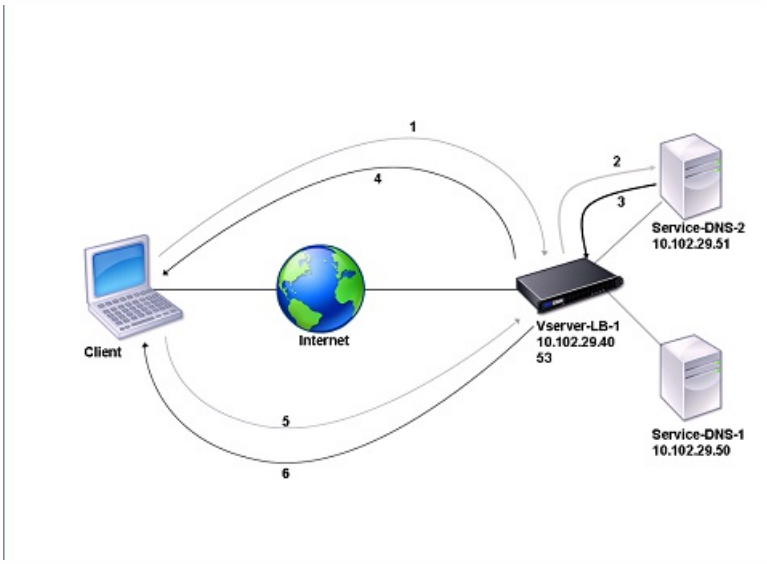
```
 Done
> show dns zone example.com
        Zone Name : example.com
        Proxy Mode : YES
Done
>
```

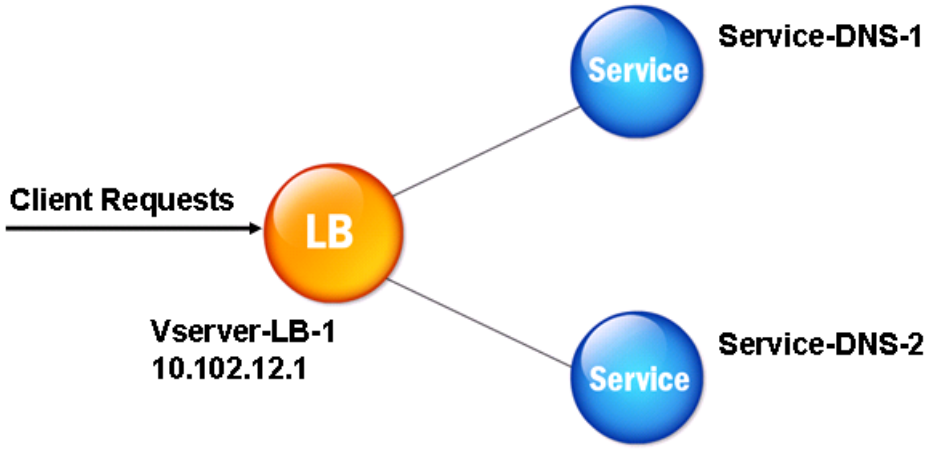
-
-





-
-
-
-
-





-
-
-
-
-
-
-
-

-
-

```
> set dns parameter -cacheRecords YES
 Done
> show dns parameter
   .
   .
   .
       Cache Records : YES
   .
   .
   .
 Done
>
```

-
-

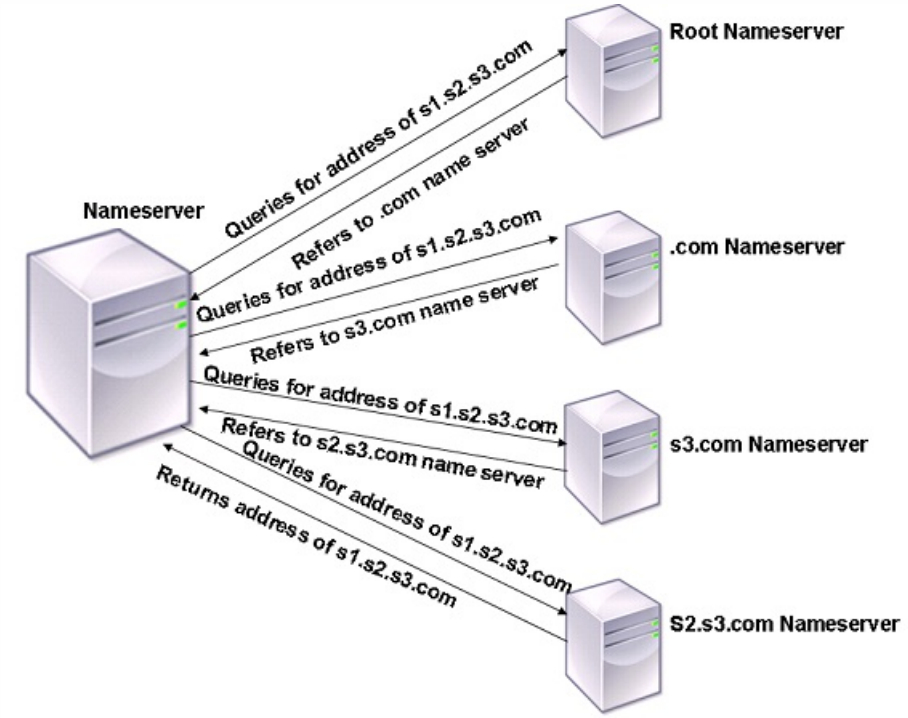
```
> set dns parameter -minTTL 1200 -maxTTL 1800
```

```
 Done
> show dns parameter
        DNS parameters:
        DNS retries: 5
        Minimum TTL: 1200           Maximum TTL: 1800
   .
   .
   .
 Done
>
```

<clientip:port>-<vserver ip:port>

-
-

```
> set dns parameter -maxPipeline 1000
 Done
> show dns parameter
      DNS parameters:
      DNS retries: 5
      .
      .
      .
      Max DNS Pipeline Requests: 1000
 Done
>
```



-
-
-
-

```
> set dns parameter -recursion ENABLED
 Done
> show dns parameter
      DNS parameters:
  .
  .
  .
      Recursive Resolution : ENABLED
  .
  .
  .
 Done
>
```

-
-

```
> set DNS parameter -retries 3
 Done
> show dns parameter
        DNS parameters:
        DNS retries: 3
   .
   .
   .
 Done
>
```

sh dns <recordtype> <domain>

-
-

```
> add dns nameServer 10.102.9.20 -local
 Done
> show dns nameServer 10.102.9.20
1)      10.102.9.20: LOCAL  -  State: UP
Done
>
```

```
> add dns nameServer dnsVirtualNS
 Done
> show dns nameServer dnsVirtualNS
1)      dnsVirtualNS  -  State: DOWN
 Done
>
```

-

-

-
-

```
> set dns parameter -nameLookupPriority DNS
 Done
> show dns parameter
   .
   .
   .
       Name lookup priority : DNS
   .
   .
   .
 Done
>
```

-
-

```
> disable dns nameServer 10.102.9.19
 Done
> show dns nameServer 10.102.9.19
1)      10.102.9.19: LOCAL  -  State: OUT OF SERVICE
 Done
>
```

-
-

```
> add dns suffix example.com
 Done
> show dns suffix example.com
1)      Suffix: example.com
 Done
>
```

-
-
-
-
-
-
-
-
-

```
> set dns parameter -dnssec ENABLED
 Done
> show dns parameter
      DNS parameters:
      DNS retries: 5
  .
  .
  .
   DNSEC Extension: ENABLED
      Max DNS Pipeline Requests: 255
 Done
>
```

bind-keygen

example.com

```
> create dns key -zoneName example.com -keyType zsk -algorithm RSASHA1 -keySize 1024 -fileNamePrefix example.com.zsk.rsasha1.1024
File Name: /nsconfig/dns/example.com.zsk.rsasha1.1024.key (public); /nsconfig/dns/example.com.zsk.rsasha1.1024.private (private); /nsconfig/dns/example.com.zsk.rsasha1
This operation may take some time, Please wait...
 Done
> create dns key -zoneName example.com -keyType ksk -algorithm RSASHA1 -keySize 4096 -fileNamePrefix example.com.ksk.rsasha1.4096
File Name: /nsconfig/dns/example.com.ksk.rsasha1.4096.key (public); /nsconfig/dns/example.com.ksk.rsasha1.4096.private (private); /nsconfig/dns/example.com.ksk.rsasha1
This operation may take some time, Please wait...
 Done
>
```

-
-
-
-
-

bind-keygen

example.com

-
-

```
> add dns key example.com.zsk example.com.zsk.rsasha1.1024.key example.com.zsk.rsasha1.1024.private
 Done
> add dns key example.com.ksk example.com.ksk.rsasha1.4096.key example.com.ksk.rsasha1.4096.private
 Done
> show dns zone example.com
        Zone Name : example.com
        Proxy Mode : NO
        Domain Name : example.com
                Record Types : NS SOA DNSKEY
        Domain Name : ns1.example.com
                Record Types : A
        Domain Name : ns2.example.com
                Record Types : A
 Done
```

>

-
-
-
-
-
-

-
-

```
> set dns key example.com.ksk -expires 30 DAYS -notificationPeriod 3 DAYS -TTL 3600
 Done
> show dns key example.com.ksk
1)      Key Name: example.com.ksk
        Expires: 30 DAYS        Notification: 3 DAYS    TTL: 3600
        Public Key File: example.com.ksk.rsasha1.4096.key
        Private Key File: example.com.ksk.rsasha1.4096.private
 Done
>
```

-
-
-

-
-
-

```
> sign dns zone example.com -keyName example.com.zsk example.com.ksk
 Done
> show dns zone example.com
```

```
        Zone Name : example.com
        Proxy Mode : NO
        Domain Name : example.com
                Record Types : NS SOA DNSKEY RRSIG NSEC
        Domain Name : ns1.example.com
                Record Types : A RRSIG NSEC
        Domain Name : ns2.example.com
                Record Types : A RRSIG
        Domain Name : ns2.example.com
                Record Types : RRSIG NSEC
Done
> save config
 Done
>
save config
```

-
-

```
> unsign dns zone example.com -keyName example.com.zsk example.com.ksk
 Done
> show dns zone example.com
        Zone Name : example.com
        Proxy Mode : NO
        Domain Name : example.com
                Record Types : NS SOA DNSKEY
        Domain Name : ns1.example.com
                Record Types : A
        Domain Name : ns2.example.com
                Record Types : A
 Done
>
```

-
-

```
> show dns nsecRec example.com
1)      Domain Name : example.com
       Next Nsec Name: ns1.example.com
       Record Types : NS SOA DNSKEY RRSIG NSEC
 Done
>
```

-
-

```
> rm dns key example.com.zsk
 Done
> show dns key example.com.zsk
ERROR: No such resource [keyName, example.com.zsk]
>
```

-
-

-
-
-

example.com.zsk1 example.com
example.com.zsk1
example.com.ksk1

-

example.com.zsk2 example.com
example.com.zsk2
example.com.zsk1 example.com.zsk2

  -

  -

-

example.com.zsk2 example.com.zsk1
example.com.zsk1 example.com.zsk1

  -
  -

-

example.com.zsk1 example.com

-

  example.com.zsk1 example.com
  example.com.zsk1
  example.com.ksk1

-

  xample.com.zsk2 example.com
  example.com.zsk2 example.com

  -

  -

  -

-

  example.com.zsk1 example.com

Offloading DNSSEC Operations to the NetScaler ADC

Nov 21, 2014

For DNS zones for which your DNS servers are authoritative, you can offload DNSSEC operations to the NetScaler ADC. In a DNSSEC offloading deployment, a DNS server sends unsigned responses. The ADC signs the response on the fly before relaying it to the client. The ADC also caches the signed response. Apart from reducing the load on the DNS servers, offloading DNSSEC operations to the ADC gives you the following benefits:

- You can sign records that the DNS servers generate programmatically. Such records cannot be signed by routine zone signing operations performed on the DNS servers.
- You can serve signed responses to clients even if you have not implemented DNSSEC on your servers.

For setting up DNSSEC offloading, you must configure a DNS load balancing virtual server, configure services that represent the DNS servers, and then bind the services to the virtual server. For information about configuring a DNS load balancing virtual server, configuring services, and binding the services to the virtual server, see Configuring a DNS Zone.

You must create a zone entity on the ADC for each DNS zone whose DNSSEC operations you want to offload. For each DNS zone, you must enable the Proxy Mode and DNSSEC Offload parameters. You can optionally configure NSEC record generation for an offloaded zone. To create a DNS zone entity for DNSSEC offloading, follow the instructions in this topic.

To complete the configuration, you must generate DNS keys for the zone, add the keys to the zone, and then sign the zone with the keys. This process is the same as for normal DNSSEC. For information about creating keys, adding keys to a zone, and signing the zone, see Domain Name System Security Extensions.

After you configure DNS offloading, you must flush the DNS cache on the ADC. Flushing the DNS cache ensures that any unsigned records in the cache are removed and subsequently replaced by signed records. For information about flushing the DNS cache, see Enabling Caching of DNS Records.

Note: DNSSEC offloading is supported on all NetScaler MPX platforms, except the NetScaler MPX 9700/10500/12500/15500 FIPS platform. The feature is also supported on NetScaler virtual appliances hosted on NetScaler SDX platforms.

At the command line, type the following commands to enable DNSSEC offloading for a zone and verify the configuration:

- add dns zone <zoneName> -proxyMode YES -dnssecOffload ENABLED [-nsec (ENABLED | DISABLED)
- show dns zone

Example

```
> add dns zone example.com -proxyMode YES -dnssecOffload ENABLED nsec ENABLED
 Done
> show dns zone example.com
  Zone Name : example.com
  Proxy Mode : YES
  DNSSEC Offload: ENABLED NSEC: ENABLED
 Done
>
```

1. Navigate to Traffic Management > DNS > Zones.
2. In the details pane, do one of the following:
   - To create a zone on the ADC, click Add.
   - To configure DNSSEC offloading for an existing zone, click the zone, and then click Open.
3. In the Create DNS Zone or Configure DNS Zone dialog box, select the Proxy Mode and DNSSEC Offload check boxes.
4. Optionally, if you want the ADC to generate NSEC records for the zone, select the NSEC check box.
5. Click OK.

Firewall Load Balancing

Jun 29, 2012

Firewall load balancing distributes traffic across multiple firewalls, providing fault tolerance and increased throughput. Firewall load balancing protects your network by:

- Dividing the load between the firewalls, which eliminates a single point of failure and allows the network to scale.
- Increasing high availability.

Configuring a NetScaler appliance for firewall load balancing is similar to configuring load balancing, with the exception that the recommended service type is ANY, recommended monitor type is PING, and the load balancing virtual server mode is set to MAC.

You can set up firewall load balancing in a sandwich, an enterprise, or multiple-firewall environment configuration. The sandwich environment is used for load balancing traffic entering the network from outside and traffic leaving the network to the internet and involves configuring two NetScaler appliances, one on each side of a set of firewalls. You configure an enterprise environment for load balancing traffic leaving the network to the internet. The enterprise environment involves configuring a single NetScaler appliance between the internal network and the firewalls that provide access to the Internet. The multiple-firewall environment is used for load balance traffic coming from another firewall. Having firewall load balancing enabled on both the sides of NetScaler improves the traffic flow in both the egress and ingress direction and ensures faster processing of the traffic. The multiple-firewall environment involves configuring a NetScaler appliance sandwiched between two firewalls.

Important: If you configure static routes on the NetScaler for the destination IP address and enable L3 mode, the NetScaler uses its routing table to route the traffic instead of sending the traffic to the load balancing vserver.
Note: For FTP to work, an additional virtual server or service should be configured on the NetScaler with IP address and port as * and 21 respectively, and the service type specified as FTP. In this case, the NetScaler manages the FTP protocol by accepting the FTP control connection, modifying the payload, and managing the data connection, all through the same firewall.
Firewall Load Balancing supports only some of the load balancing methods supported on the NetScaler. Also, you can configure only a few types of persistence and monitors.

The following load balancing methods are supported for firewall load balancing.

- Least Connections
- Round Robin
- Least Packets
- Least Bandwidth
- Source IP Hash
- Destination IP Hash
- Source IP Destination IP Hash
- Source IP Source Port hash
- Least Response Time Method (LRTM)
- Custom Load

Only SOURCEIP, DESTIP, and SOURCEIPDESTIP based persistence are supported for firewall load balancing.

Only PING and transparent monitors are supported in firewall load balancing. You can bind a PING monitor (default) to the backend service that represents the firewall. If a firewall is configured not to respond to ping packets, you can configure transparent monitors to monitor hosts on the trusted side through individual firewalls.

Sandwich Environment

Apr 08, 2013

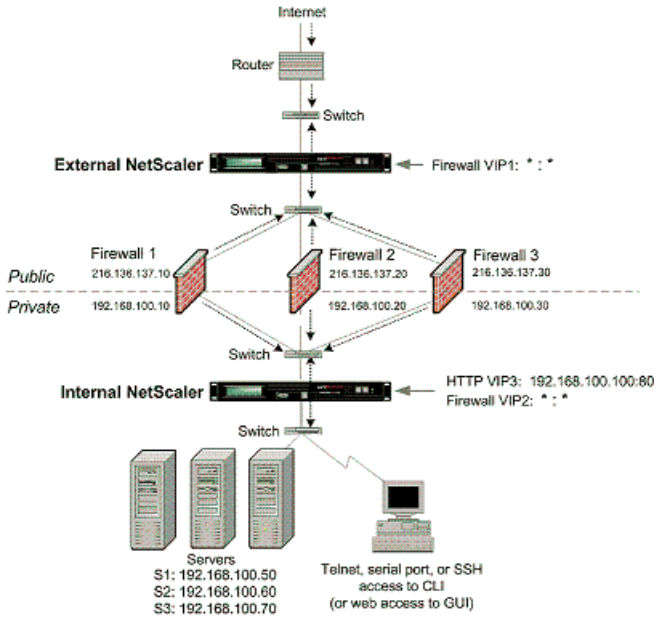
A NetScaler deployment in a sandwich mode is capable of load balancing network traffic through firewalls in both directions: ingress (traffic entering the network from the outside, such as the internet) and egress (traffic leaving the network to the internet).

In this setup, a NetScaler is located on each side of a set of firewalls. The NetScaler placed between the firewalls and the Internet, called the external NetScaler that handles ingress traffic selects the best firewall, based on the configured method. The NetScaler between the firewalls and the private network, called the internal NetScaler tracks the firewall from which the initial packet for a session is received. It then makes sure that all subsequent packets for that session are sent to the same firewall.

The internal NetScaler can be configured as a regular traffic manager to load balance traffic across the private network servers. This configuration also allows traffic originating from the private network (egress) to be load balanced across the firewalls.

The following diagram shows the sandwich firewall load balancing environment.

Figure 1. Firewall Load Balancing (Sandwich)



The service type ANY configures the NetScaler to accept all traffic.

To avail benefits related to HTTP and TCP, configure the service and virtual server with type HTTP or TCP. For FTP to work, configure the service with type FTP.

Updated: 2015-05-22

Perform the following tasks to configure the external NetScaler in a sandwich environment

- Enable the load balancing feature.

- Configure a wildcard service for each firewall.
- Configure a monitor for each wildcard service.
- Configure a wildcard virtual server for traffic coming from the Internet.
- Configure the virtual server in MAC rewrite mode.
- Bind services to the wildcard virtual server.
- Save and Verify the Configuration.

Enable the load balancing feature

At the command prompt, type the following command to enable load balancing and verify the configuration:

- enable ns feature LB
- show ns feature

Example

```
> enable ns feature LoadBalancing
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection         SP              ON
 3)    Load Balancing           LB              ON
 .
 .
 .
 24)   NetScaler Push           push            OFF
 Done
```

Configure a wildcard service for each firewall

At the command prompt, type:

add service <name> <serverName> ANY *

Example

```
add service Service-HTTP-1 10.102.29.5 ANY *
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:
   - Service Name—name
   - Server—serverName
4. In Protocol, select ANY, and in Port, select *.

5. Click Create, and then click Close. The service you created appears in the Services pane.

Configure a monitor for each wildcard service

A PING monitor is bound by default to the service. You will need to configure a transparent monitor to monitor hosts on the trusted side through individual firewalls. You can then bind the transparent monitor to services. The default PING monitor monitors the connectivity only between the NetScaler appliance and the upstream device. The transparent monitor monitors all the devices existing in the path from the appliance to the device that owns the destination IP address specified in the monitor. If a transparent monitor is not configured and the status of the firewall is UP but one of the next hop devices from that firewall is down, the appliance includes the firewall while performing load balancing and forwards the packet to the firewall. However, the packet is not delivered to the final destination because one of the next hop devices is down. By binding a transparent monitor, if any of the devices (including the firewall) are down, the service is marked as DOWN and the firewall is not included when the appliance performs firewall load balancing.

Binding a transparent monitor will override the PING monitor. To configure a PING monitor in addition to a transparent monitor, after you create and bind a transparent monitor, you need to bind a PING monitor to the service.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- add lb monitor <monitorName> <type> [-destIP <ip_addr|ipv6_addr|*>] [-transparent (YES | NO)]
- bind lb monitor <monitorName> <serviceName>

Example

```
add monitor monitor-HTTP-1 HTTP -destip 10.10.10.11 -transparent YES
bind monitor monitor-HTTP-1 fw-svc1
To bind a PING monitor, type the following command:
bind monitor PING fw-svc1
```

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify values for the following parameters as shown:
   - Name*
   - Type*
   - Destination IP
   - Transparent

   * A required parameter
4. Click Create, and then click Close. In the Monitors pane, select the monitor that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configure a wildcard virtual server for traffic coming from the Internet

At the command prompt, type:

add lb vserver <name> ANY * *

Example

```
add lb vserver Vserver-LB-1 ANY * *
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters as shown:
   - Name—name
4. In Protocol, select ANY, and in IP Address and Port, select *.
5. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Configure the virtual server in MAC rewrite mode

At the command prompt, type:

set lb vserver <name>@ -m <RedirectionMode>

Example

```
set lb vserver Vserver-LB-1 -m MAC
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode (for example, Vserver-LB-1), and then click Open.
3. On the Advanced tab, under Redirection Mode, click MAC-Based.
4. Click OK.

Bind services to the wildcard virtual server

At the command prompt, type:

bind lb vserver <name> <serviceName>

Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the service.
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server.
5. Click OK.

Note: You can bind a service to multiple virtual servers.

Save and Verify the Configuration

When you've finished the configuration tasks, be sure to save the configuration. You should also check to make sure that the settings are correct.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- save ns config
- show vserver

Example

```
save config
sh lb vserver FWLBVIP1
FWLBVIP1 (*:*) - ANY    Type: ADDRESS
        State: UP
        Last state change was at Mon Jun 14 06:40:14 2010
        Time since last state change: 0 days, 00:00:11.240
        Effective State: UP  ARP:DISABLED
        Client Idle Timeout: 120 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        No. of Bound Services :  2 (Total)       2 (Active)
        Configured Method: SRCIPDESTIPHASH
        Mode: MAC
        Persistence: NONE
        Connection Failover: DISABLED

1) fw_svc_1 (10.102.29.251: *) - ANY State: UP   Weight: 1
2) fw_svc_2 (10.102.29.18: *) - ANY State: UP    Weight: 1
 Done
show service fw-svc1
        fw-svc1 (10.102.29.251:*) - ANY
        State: DOWN
        Last state change was at Thu Jul  8 10:04:50 2010
        Time since last state change: 0 days, 00:00:38.120
        Server Name: 10.102.29.251
        Server ID : 0   Monitor Threshold : 0
        Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
        Use Source IP: NO
        Client Keepalive(CKA): NO
        Access Down Service: NO
        TCP Buffering(TCPB): YES
        HTTP Compression(CMP): NO
        Idle timeout: Client: 120 sec   Server: 120 sec
        Client IP: DISABLED
        Cacheable: NO
        SC: OFF
        SP: OFF
        Down state flush: ENABLED
```

```
1)    Monitor Name: monitor-HTTP-1
          State: DOWN    Weight: 1
          Probes: 5      Failed [Total: 5 Current: 5]
          Last response: Failure - Time out during TCP connection establishment stage
          Response Time: 2000.0 millisec
2)    Monitor Name: ping
          State: UP      Weight: 1
          Probes: 3      Failed [Total: 0 Current: 0]
          Last response: Success - ICMP echo reply received.
          Response Time: 1.415 millisec
 Done
```

1. In the details pane, click Save.
2. In the Save Config dialog box, click Yes.
3. Navigate to Traffic Management > Load Balancing > Virtual Servers.
4. In the details pane, select the virtual server that you created in step 5 and verify that the settings displayed in the Details pane are correct.
5. Navigate to Traffic Management > Load Balancing > Virtual Servers.
6. In the details pane, select the service that you created in step 5 and verify that the settings displayed in the Details pane are correct.

Updated: 2015-06-04

Perform the following tasks to configure the internal NetScaler in a sandwich environment

For traffic from the server (egress)

- Enable the load balancing feature.
- Configure a wildcard service for each firewall.
- Configure a monitor for each wildcard service.
- Configure a wildcard virtual server to load balance the traffic sent to the firewalls.
- Configure the virtual server in MAC rewrite mode.
- Bind firewall services to the wildcard virtual server.

For traffic across private network servers

- Configure a service for each virtual server .
- Configure a monitor for each service.
- Configure an HTTP virtual server to balance traffic sent to the servers.
- Bind HTTP services to the HTTP virtual server .
- Save and Verify the Configuration.

Enable the load balancing feature

You can configure load balancing entities such as services and virtual servers when the load balancing feature is disabled, but they will not function until you enable the feature.

At the command prompt, type the following command to enable load balancing and verify the configuration:

- enable ns feature LB
- show ns feature

Example

```
> enable ns feature LoadBalancing
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection          SP              ON
 3)    Load Balancing            LB              ON
 .
 .
 .
 24)   NetScaler Push            push            OFF
Done
```

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Load Balancing check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes.

Configure a wildcard service for each firewall

At the command prompt, type:

add service <name> <serverName> ANY *

Example

```
add service Service-HTTP-1 10.102.29.5 ANY *
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:
   - Service Name—name
   - Server—serverName
4. In Protocol, select ANY, and in Port, select *.
5. Click Create, and then click Close. The service you created appears in the Services pane.

Configure a monitor for each wildcard service

A PING monitor is bound by default to the service. You will need to configure a transparent monitor to monitor hosts on the trusted side through individual firewalls. You can then bind the transparent monitor to services. The default PING monitor monitors the connectivity only between the NetScaler appliance and the upstream device. The transparent monitor monitors all the devices existing in the path from the appliance to the device that owns the destination IP address specified in the monitor. If a transparent monitor is not configured and the status of the firewall is UP but one of the next hop devices from that firewall is down, the appliance includes the firewall while performing load balancing and forwards the packet to the firewall. However, the packet is not delivered to the final destination because one of the next hop devices is down. By binding a transparent monitor, if any of the devices (including the firewall) are down, the service is marked as DOWN and the firewall is not included when the appliance performs firewall load balancing.

Binding a transparent monitor will override the PING monitor. To configure a PING monitor in addition to a transparent monitor, after you create and bind a transparent monitor, you need to bind a PING monitor to the service.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- add lb monitor <monitorName> <type> [-destIP <ip_addr|ipv6_addr|*>] [-transparent (YES | NO)]
- bind lb monitor <monitorName> <serviceName>

Example

```
add monitor monitor-HTTP-1 HTTP -destip 10.10.10.11 -transparent YES
bind monitor monitor-HTTP-1 fw-svc1
```

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify values for the following parameters as shown:
   - Name*
   - Type*
   - Destination IP
   - Transparent

   * A required parameter
4. Click Create, and then click Close. In the Monitors pane, select the monitor that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configure a wildcard virtual server to load balance the traffic sent to the firewalls

At the command prompt, type:

add lb vserver <name> ANY * *

Example

```
add lb vserver Vserver-LB-1 ANY * *
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters as shown:
   - Name—name
4. In Protocol, select ANY, and in IP Address and Port, select *.
5. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Configure the virtual server in MAC rewrite mode

At the command prompt, type:

set lb vserver <name>@ -m <RedirectionMode>

Example

```
set lb vserver Vserver-LB-1 -m MAC
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode (for example, Vserver-LB-1), and then click Open.
3. On the Advanced tab, under Redirection Mode, click MAC-Based.
4. Click OK.

Bind firewall services to the wildcard virtual server

At the command prompt, type:

bind lb vserver <name> <serviceName>

Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the service.
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server.
5. Click OK.

Note: You can bind a service to multiple virtual servers.

Configure a service for each virtual server

At the command prompt, type:

add service <name> <serverName> HTTP <port>

Example

```
add service Service-HTTP-1 10.102.29.5 HTTP 80
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:
   - Service Name—name
   - Server—serverName
   - Port—port
4. In Protocol, specify HTTP. Under Available Monitors, select HTTP.
5. Click Create, and then click Close. The service you created appears in the Services pane.

Configure a monitor for each service

At the command prompt, type:

bind lb monitor <monitorName> <ServiceName>

Example

```
bind mon monitor-HTTP-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to bind the monitor (for example, Service-HTTP-1), and then click Open.
3. On the Monitors tab, in the Available list box, select the monitor you want to bind the service (for example, monitor-HTTP-1), and then click Add.
4. In the Configured box, click OK.

Configure an HTTP virtual server to balance traffic sent to the servers

At the command prompt, type:

add lb vserver <name> HTTP <ip> <port>

Example

```
add lb vserver Vserver-LB-1 HTTP 10.102.29.60 80
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters as shown:

- Name—name
- IP Address—IPAddress
  Note: If the virtual server uses IPv6, select the IPv6 check box and enter the address in IPv6 format (for example, 1000:0000:0000:0000:0005:0600:700a:888b).
- Port—port

4. Under Protocol, select HTTP.
5. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Bind HTTP services to the HTTP virtual server

At the command prompt, type:

bind lb vserver <name> <serviceName>

Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the service.
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server.
5. Click OK.

Note: You can bind a service to multiple virtual servers.

Save and Verify the Configuration

When you've finished the configuration tasks, be sure to save the configuration. You should also check to make sure that the settings are correct.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- save ns config
- show vserver

Example

```
save config
show lb vserver FWLBVIP2
    FWLBVIP2 (*:*) - ANY    Type: ADDRESS
    State: UP
    Last state change was at Mon Jun 14 07:22:54 2010
    Time since last state change: 0 days, 00:00:32.760
    Effective State: UP
    Client Idle Timeout: 120 sec
    Down state flush: ENABLED
```

```
        Disable Primary Vserver On Down : DISABLED
        No. of Bound Services :  2 (Total)       2 (Active)
        Configured Method: LEASTCONNECTION
        Current Method: Round Robin, Reason: A new service is bound
        Mode: MAC
        Persistence: NONE
        Connection Failover: DISABLED

1) fw-int-svc1 (10.102.29.5: *) - ANY State: UP Weight: 1
2) fw-int-svc2 (10.102.29.9: *) - ANY State: UP Weight: 1
 Done
show service fw-int-svc1
        fw-int-svc1 (10.102.29.5:*) - ANY
        State: DOWN
        Last state change was at Thu Jul  8 14:44:51 2010
        Time since last state change: 0 days, 00:01:50.240
        Server Name: 10.102.29.5
        Server ID : 0   Monitor Threshold : 0
        Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
        Use Source IP: NO
        Client Keepalive(CKA): NO
        Access Down Service: NO
        TCP Buffering(TCPB): NO
        HTTP Compression(CMP): NO
        Idle timeout: Client: 120 sec   Server: 120 sec
        Client IP: DISABLED
        Cacheable: NO
        SC: OFF
        SP: OFF
        Down state flush: ENABLED

1)      Monitor Name: monitor-HTTP-1
              State: DOWN     Weight: 1
              Probes: 9       Failed [Total: 9 Current: 9]
              Last response: Failure - Time out during TCP connection establishment stage
              Response Time: 2000.0 millisec
2)      Monitor Name: ping
              State: UP       Weight: 1
              Probes: 3       Failed [Total: 0 Current: 0]
              Last response: Success - ICMP echo reply received.
              Response Time: 1.275 millisec
 Done
```

1. In the details pane, click Save.
2. In the Save Config dialog box, click Yes.
3. Navigate to Traffic Management > Load Balancing > Virtual Servers.
4. In the details pane, select the virtual server that you created in step 5 and verify that the settings displayed in the Details pane are correct.

5. Navigate to Traffic Management > Load Balancing > Services.
6. In the details pane, select the service that you created in step 5 and verify that the settings displayed in the Details pane are correct.

After the configuration is up and running, you should view the statistics for each service and virtual server to check for possible problems.

Viewing the Statistics of a Virtual Server

Updated: 2013-09-05

To evaluate the performance of virtual servers or to troubleshoot problems, you can display details of the virtual servers configured on the NetScaler appliance. You can display a summary of statistics for all the virtual servers, or you can specify the name of a virtual server to display the statistics only for that virtual server. You can display the following details:

- Name
- IP address
- Port
- Protocol
- State of the virtual server
- Rate of requests received
- Rate of hits

To display a summary of the statistics for all the virtual servers currently configured on the NetScaler, or for a single virtual server, at the command prompt, type:

stat lb vserver [-detail] [<name>]

Example

```
>stat lb vserver -detail
Virtual Server(s) Summary
                vsvrIP  port    Protocol      State   Req/s   Hits/s
One                  *    80        HTTP         UP      5/s      0/s
Two                  *     0         TCP       DOWN      0/s      0/s
Three                *  2598         TCP       DOWN      0/s      0/s
dnsVirtualNS    10.102.29.90    53         DNS       DOWN      0/s      0/s
BRVSERV          10.10.1.1    80        HTTP       DOWN      0/s      0/s
LBVIP         10.102.29.66    80        HTTP         UP      0/s      0/s
 Done
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. If you want to display the statistics for only one virtual server, in the details pane, select the virtual server whose statistics you want to display.
3. In the details pane, click Statistics.

Viewing the Statistics of a Service

Updated: 2013-08-28

You can view the rate of requests, responses, request bytes, response bytes, current client connections, requests in surge queue, current server connections, and so forth using the service statistics.

At the command prompt, type:

stat service <name>

Example

```
stat service Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service whose statistics you want to view (for example, Service-HTTP-1).
3. Click Statistics. The statistics appear in a new window.

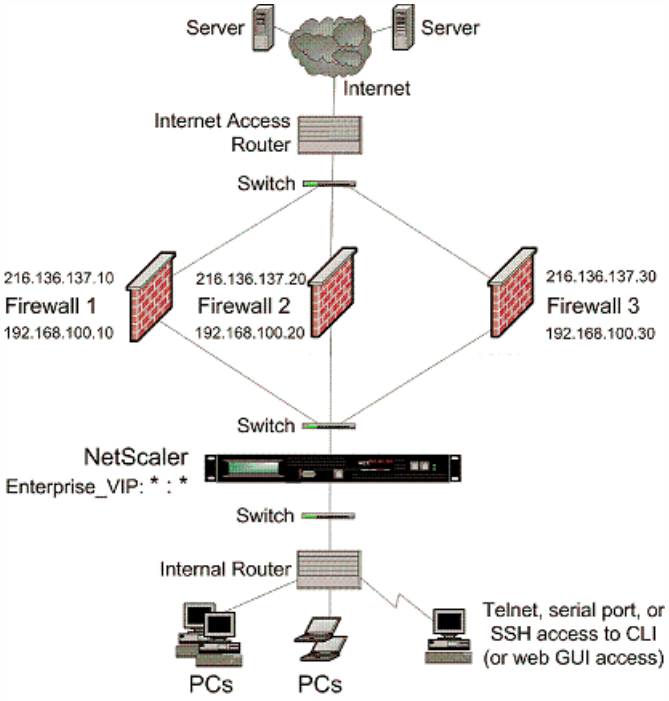
Enterprise Environment

Mar 22, 2012

In the enterprise setup, the NetScaler is placed between the firewalls connecting to the public Internet and the internal private network and handles egress traffic. The NetScaler selects the best firewall based on the configured load balancing policy.

The following diagram shows the enterprise firewall load balancing environment

Figure 1. Firewall Load Balancing (Enterprise)



The service type ANY configures the NetScaler to accept all traffic.

To avail benefits related to HTTP and TCP, configure the service and vserver with type HTTP or TCP. For FTP to work, configure the service with type FTP.

Updated: 2013-11-08

Perform the following tasks to configure a NetScaler in an enterprise environment.

For traffic from the server (egress)

- Enable the load balancing feature.
- Configure a wildcard service for each firewall.
- Configure a monitor for each wildcard service.
- Configure a wildcard virtual server to load balance the traffic sent to the firewalls .
- Configure the virtual server in MAC rewrite mode.
- Bind firewall services to the wildcard virtual server.

For traffic across private network servers

- Configure a service for each virtual server .
- Configure a monitor for each service.
- Configure an HTTP virtual server to balance traffic sent to the servers.
- Bind HTTP services to the HTTP virtual server .
- Save and Verify the Configuration.

Enable the load balancing feature

You can configure load balancing entities such as services and virtual servers when the load balancing feature is disabled, but they will not function until you enable the feature.

At the command prompt, type the following command to enable load balancing and verify the configuration:

- enable ns feature LB
- show ns feature

Example

```
> enable ns feature LoadBalancing
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection          SP              ON
 3)    Load Balancing            LB              ON
 .
 .
 .
 24)   NetScaler Push            push            OFF
Done
```

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Load Balancing check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes.

Configure a wildcard service for each firewall

At the command prompt, type:

add service <name> <serverName> ANY *

Example

```
add service Service-HTTP-1 10.102.29.5 ANY *
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:
   - Service Name—name
   - Server—serverName
4. In Protocol, select ANY, and in Port, select *.
5. Click Create, and then click Close. The service you created appears in the Services pane.

Configure a monitor for each wildcard service

A PING monitor is bound by default to the service. You will need to configure a transparent monitor to monitor hosts on the trusted side through individual firewalls. You can then bind the transparent monitor to services. The default PING monitor monitors the connectivity only between the NetScaler appliance and the upstream device. The transparent monitor monitors all the devices existing in the path from the appliance to the device that owns the destination IP address specified in the monitor. If a transparent monitor is not configured and the status of the firewall is UP but one of the next hop devices from that firewall is down, the appliance includes the firewall while performing load balancing and forwards the packet to the firewall. However, the packet is not delivered to the final destination because one of the next hop devices is down. By binding a transparent monitor, if any of the devices (including the firewall) are down, the service is marked as DOWN and the firewall is not included when the appliance performs firewall load balancing.

Binding a transparent monitor will override the PING monitor. To configure a PING monitor in addition to a transparent monitor, after you create and bind a transparent monitor, you need to bind a PING monitor to the service.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- add lb monitor <monitorName> <type> [-destIP <ip_addr|ipv6_addr|*>] [-transparent (YES | NO)]
- bind lb monitor <monitorName> <serviceName>

Example

```
add monitor monitor-HTTP-1 HTTP -destip 10.10.10.11 -transparent YES
bind monitor monitor-HTTP-1 fw-svc1
```

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify values as shown:

   - Name*
   - Type*—type
   - Destination IP
   - Transparent

   * A required parameter

4. Click Create, and then click Close. In the Monitors pane, select the monitor that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configure a wildcard virtual server to load balance the traffic sent to the firewalls

At the command prompt, type:

add lb vserver <name> ANY * *

Example

```
add lb vserver Vserver-LB-1 ANY * *
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters as shown:
   - Name—name
4. In Protocol, select ANY, and in IP Address and Port, select *.
5. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Configure the virtual server in MAC rewrite mode

At the command prompt, type:

set lb vserver <name>@ -m <RedirectionMode>

Example

```
set lb vserver Vserver-LB-1 -m MAC
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode (for example, Vserver-LB-1), and then click Open.
3. On the Advanced tab, under Redirection Mode, click MAC-Based.
4. Click OK.

Bind firewall services to the wildcard virtual server

At the command prompt, type:

bind lb vserver <name> <serviceName>

Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the service.
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server.
5. Click OK.

Note: You can bind a service to multiple virtual servers.

Configure a service for each virtual server

At the command prompt, type:

add service <name> <serverName> HTTP <port>

Example

```
add service Service-HTTP-1 10.102.29.5 HTTP 80
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters as shown:
   - Service Name—name
   - Server—serverName
   - Port—port
4. In Protocol, specify HTTP. Under Available Monitors, select HTTP.
5. Click Create, and then click Close. The service you created appears in the Services pane.

Configure a monitor for each service

At the command prompt, type:

bind lb monitor <monitorName> <ServiceName>

Example

```
bind mon monitor-HTTP-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to bind the monitor (for example, Service-HTTP-1), and then click Open.
3. On the Monitors tab, in the Available list box, select the monitor you want to bind the service (for example, monitor-HTTP-1), and then click Add.
4. In the Configured box, click OK.

Configure an HTTP virtual server to balance traffic sent to the servers

At the command prompt, type:

add lb vserver <name> HTTP <ip> <port>
Example

```
add lb vserver Vserver-LB-1 HTTP 10.102.29.60 80
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters as shown:
   - Name—name
   - IP Address—IPAddress
     Note: If the virtual server uses IPv6, select the IPv6 check box and enter the address in IPv6 format (for example, **1000:0000:0000:0000:0005:0600:700a:888b**).
   - Port—port
4. Under Protocol, select HTTP.
5. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Bind HTTP services to the HTTP virtual server

At the command prompt, type:

bind lb vserver <name> <serviceName>
Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the service.
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server.
5. Click OK.

Note: You can bind a service to multiple virtual servers.

Save and Verify the Configuration

When you've finished the configuration tasks, be sure to save the configuration. You should also check to make sure that the settings are correct.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- save ns config
- show vserver

Example

```
save config
show lb vserver FWLBVIP2
        FWLBVIP2 (*:*) - ANY    Type: ADDRESS
        State: UP
        Last state change was at Mon Jun 14 07:22:54 2010
        Time since last state change: 0 days, 00:00:32.760
        Effective State: UP
        Client Idle Timeout: 120 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        No. of Bound Services :  2 (Total)       2 (Active)
        Configured Method: LEASTCONNECTION
        Current Method: Round Robin, Reason: A new service is bound
        Mode: MAC
        Persistence: NONE
        Connection Failover: DISABLED

1) fw-int-svc1 (10.102.29.5: *) - ANY State: UP Weight: 1
2) fw-int-svc2 (10.102.29.9: *) - ANY State: UP Weight: 1
 Done
show service fw-int-svc1
        fw-int-svc1 (10.102.29.5:*) - ANY
        State: DOWN
        Last state change was at Thu Jul  8 14:44:51 2010
        Time since last state change: 0 days, 00:01:50.240
        Server Name: 10.102.29.5
        Server ID : 0   Monitor Threshold : 0
        Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
        Use Source IP: NO
        Client Keepalive(CKA): NO
        Access Down Service: NO
        TCP Buffering(TCPB): NO
        HTTP Compression(CMP): NO
        Idle timeout: Client: 120 sec   Server: 120 sec
        Client IP: DISABLED
        Cacheable: NO
        SC: OFF
        SP: OFF
        Down state flush: ENABLED

1)      Monitor Name: monitor-HTTP-1
                State: DOWN     Weight: 1
                Probes: 9       Failed [Total: 9 Current: 9]
                Last response: Failure - Time out during TCP connection establishment stage
                Response Time: 2000.0 millisec
2)      Monitor Name: ping
                State: UP       Weight: 1
```

```
        Probes: 3      Failed [Total: 0 Current: 0]
        Last response: Success - ICMP echo reply received.
        Response Time: 1.275 millisec
Done
```

1. In the details pane, click Save.
2. In the Save Config dialog box, click Yes.
3. Navigate to Traffic Management > Load Balancing > Virtual Servers.
4. In the details pane, select the virtual server that you created in step 5 and verify that the settings displayed in the Details pane are correct.
5. Navigate to Traffic Management > Load Balancing > Services.
6. In the details pane, select the service that you created in step 5 and verify that the settings displayed in the Details pane are correct.

After the configuration is up and running, you should view the statistics for each service and virtual server to check for possible problems.

Viewing the Statistics of a Virtual Server

Updated: 2013-09-05

To evaluate the performance of virtual servers or to troubleshoot problems, you can display details of the virtual servers configured on the NetScaler appliance. You can display a summary of statistics for all the virtual servers, or you can specify the name of a virtual server to display the statistics only for that virtual server. You can display the following details:

- Name
- IP address
- Port
- Protocol
- State of the virtual server
- Rate of requests received
- Rate of hits

To display a summary of the statistics for all the virtual servers currently configured on the NetScaler, or for a single virtual server, at the command prompt, type:

stat lb vserver [-detail] [<name>]

Example

```
>stat lb vserver -detail
Virtual Server(s) Summary
                vsvrIP  port    Protocol      State   Req/s   Hits/s
One                  *    80        HTTP         UP      5/s      0/s
Two                  *     0         TCP       DOWN      0/s      0/s
Three                *  2598         TCP       DOWN      0/s      0/s
dnsVirtualNS    10.102.29.90    53         DNS       DOWN      0/s      0/s
```

```
BRVSERV          10.10.1.1    80       HTTP        DOWN      0/s      0/s
LBVIP         10.102.29.66    80       HTTP          UP      0/s      0/s
 Done
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. If you want to display the statistics for only one virtual server, in the details pane, select the virtual server whose statistics you want to display.
3. In the details pane, click Statistics.

Viewing the Statistics of a Service

Updated: 2013-08-28

You can view the rate of requests, responses, request bytes, response bytes, current client connections, requests in surge queue, current server connections, and so forth using the service statistics.

At the command prompt, type:

stat service <name>
Example

stat service Service-HTTP-1

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service whose statistics you want to view (for example, Service-HTTP-1).
3. Click Statistics. The statistics appear in a new window.

Multiple-Firewall Environment

Sep 06, 2012

Note: This feature is available in NetScaler release 9.3.e and 10.

In a multiple-firewall environment, the NetScaler appliance is placed between two sets of firewalls, the external set connecting to the public Internet, and the internal set connecting to the internal private network. The external set typically handles the egress traffic. These firewalls mainly implement access control lists to allow or deny access to external resources. The internal set typically handles the ingress traffic. These firewalls implement security to safeguard the intranet from malicious attacks apart from load-balancing the ingress traffic. The multiple-firewall environment allows you to load-balance traffic coming from another firewall. By default, the traffic coming from a firewall is not load balanced on the other firewall across a NetScaler. Having firewall load balancing enabled on both the sides of NetScaler improves the traffic flow in both the egress and ingress direction and ensures faster processing of the traffic.

Figure 1 shows a multiple-firewall load balancing environment

Figure 1. Firewall Load Balancing (multiple-firewall)

With a configuration like the one shown in Figure 1, you can configure the NetScaler to load balance the traffic through the an internal firewall even if it is load balanced by an external firewall. For example, with this feature configured, the traffic coming from the external firewalls (firewalls 1, 2, and 3) is load balanced on the internal firewalls (firewalls 4, 5, and 6) and vice versa.

Firewall load balancing is supported only for MAC mode LB virtual server.

The service type ANY configures the NetScaler to accept all traffic.

To avail benefits related to HTTP and TCP, configure the service and virtual server with type HTTP or TCP. For FTP to work, configure the service with type FTP.

Updated: 2015-05-18

To configure a NetScaler appliance in a multiple-firewall environment, you have to enable the load balancing feature, configure a virtual server to load balance the egress traffic across the external firewalls, configure a virtual server to load balance the ingress traffic across the internal firewalls, and enable firewall load balancing on the NetScaler. To configure a virtual server to load balance traffic across a firewall in the multiple-firewall environment, you need to:

1. Configure a wildcard service for each firewall
2. Configure a monitor for each wildcard service
3. Configure a wildcard virtual server to load balance the traffic sent to the firewalls
4. Configure the virtual server in MAC rewrite mode
5. Bind firewall services to the wildcard virtual server

Enabling the load balancing feature

To configure and implement load balancing entities such as services and virtual servers, you need to enable the load balancing feature on the NetScaler device.

At the command prompt, type the following command to enable load balancing and verify the configuration:

- enable ns feature <featureName>
- show ns feature

Example

```
enable ns feature LoadBalancing
Done
show ns feature
Feature Acronym Status
------- ------- ------
1) Web Logging WL OFF
2) Surge Protection SP ON
3) Load Balancing LB ON
.
.
.
24) NetScaler Push push OFF
Done
```

1. In the navigation pane, expand System, and then click Settings.
2. In the Settings pane, under Modes and Features, click Change basic features.

3. In the Configure Basic Features dialog box, select the Load Balancing check box, and then click Ok.

Configuring a wildcard service for each firewall

To accept traffic from all the protocols, you need to configure wildcard service for each firewall by specifying support for all the protocols and ports.

At the command prompt, type the following command to configure support for all the protocols and ports:

add service <name>@ <serverName> <serviceType> <port_number>

Example

```
add service fw-svc1 10.102.29.5 ANY *
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Services dialog box, specify values for the following parameters as shown:

   - Service Name—name
   - Server—serverName

   * A required parameter

4. In Protocol, select Any and in Port, select *.
5. Click Create, and then click Close. The service you created appears in the Services pane.

Configuring a monitor for each service

A PING monitor is bound by default to the service. You will need to configure a transparent monitor to monitor hosts on the trusted side through individual firewalls. You can then bind the transparent monitor to services. The default PING monitor monitors the connectivity only between the NetScaler appliance and the upstream device. The transparent monitor monitors all the devices existing in the path from the appliance to the device that owns the destination IP address specified in the monitor. If a transparent monitor is not configured and the status of the firewall is UP but one of the next hop devices from that firewall is down, the appliance includes the firewall while performing load balancing and forwards the packet to the firewall. However, the packet is not delivered to the final destination because one of the next hop devices is down. By binding a transparent monitor, if any of the devices (including the firewall) are down, the service is marked as DOWN and the firewall is not included when the appliance performs firewall load balancing.

Binding a transparent monitor will override the PING monitor. To configure a PING monitor in addition to a transparent monitor, after you create and bind a transparent monitor, you need to bind a PING monitor to the service.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- add lb monitor <monitorName> <type> [-destIP <ip_addr|ipv6_addr|*>] [-transparent (YES | NO)]
- bind lb monitor <monitorName> <serviceName>

Example

```
add monitor monitor-HTTP-1 HTTP -destip 10.10.10.11 -transparent YES
```

bind monitor monitor-HTTP-1 fw-svc1

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify values for the following parameters as shown:

   - Name*
   - Type*—type
   - Destination IP
   - Transparent

   * A required parameter

4. Click Create, and then click Close. In the Monitors pane, select the monitor that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configuring a virtual server to load balance the traffic sent to the firewalls

To load balance any kind of traffic, you need to configure a wildcard virtual server specifying the protocol and port as any value.

At the command prompt, type the following command:

add lb vserver <name>@ <serviceType> <IPAddress> <port_number>
Example

add lb vserver Vserver-LB-1 ANY * *

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In Protocol, select Any, and in IP Address and Port, select *.
4. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Configuring the virtual server to MAC rewrite mode

To configure the virtual server to use MAC address for forwarding the incoming traffic, you need to enable the MAC rewrite mode.

At the command prompt, type the following command:

set lb vserver <name>@ -m <RedirectionMode>
Example

set lb vserver Vserver-LB-1 -m MAC

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode (for example, Vserver-

LB1), and then click Open.
3. On the Advanced tab, under the Redirection Mode mode, click Open.
4. Click Ok.

Binding firewall services to the virtual server

To access a service on NetScaler, you need to bind it to a wildcard virtual server.

At the command prompt, type the following command:

bind lb vserver <name>@ <serviceName>
Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode (for example, Vserver-LB1), and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server(for example, Service-HTTP-1).
4. Click Ok.

Configuring the multiple-firewall load balancing on the NetScaler Appliance

To load balance traffic on both the sides of a NetScaler using firewall load balancing, you need to enable mulitpl-firewall load balancing by using the vServerSpecificMac parameter.

At the command prompt, type the following command:

set lb parameter -vServerSpecificMac <status>
Example

```
set lb parameter -vServerSpecificMac ENABLED
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode (for example, Configure Load Balancing parameters).
3. In the Set Load Balancing Parameters dialog box, select the Virtual Server Specific MAC check box.
4. Click Ok.

Saving and Verifying the Configuration

When you've finished the configuration tasks, be sure to save the configuration. You should also check to make sure that the settings are correct.

At the command prompt, type the following commands to configure a transparent monitor and verify the configuration:

- save ns config
- show vserver

Example

```
save config
show lb vserver FWLBVIP2
        FWLBVIP2 (*:*) - ANY    Type: ADDRESS
        State: UP
        Last state change was at Mon Jun 14 07:22:54 2010
        Time since last state change: 0 days, 00:00:32.760
        Effective State: UP
        Client Idle Timeout: 120 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        No. of Bound Services :  2 (Total)      2 (Active)
        Configured Method: LEASTCONNECTION
        Current Method: Round Robin, Reason: A new service is bound
        Mode: MAC
        Persistence: NONE
        Connection Failover: DISABLED

1) fw-int-svc1 (10.102.29.5: *) - ANY State: UP Weight: 1
2) fw-int-svc2 (10.102.29.9: *) - ANY State: UP Weight: 1
 Done
show service fw-int-svc1
        fw-int-svc1 (10.102.29.5:*) - ANY
        State: DOWN
        Last state change was at Thu Jul  8 14:44:51 2010
        Time since last state change: 0 days, 00:01:50.240
        Server Name: 10.102.29.5
        Server ID : 0   Monitor Threshold : 0
        Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
        Use Source IP: NO
        Client Keepalive(CKA): NO
        Access Down Service: NO
        TCP Buffering(TCPB): NO
        HTTP Compression(CMP): NO
        Idle timeout: Client: 120 sec   Server: 120 sec
        Client IP: DISABLED
        Cacheable: NO
        SC: OFF
        SP: OFF
        Down state flush: ENABLED

1)      Monitor Name: monitor-HTTP-1
              State: DOWN     Weight: 1
              Probes: 9       Failed [Total: 9 Current: 9]
              Last response: Failure - Time out during TCP connection establishment stage
```



```
        Response Time: 2000.0 millisec
2)      Monitor Name: ping
        State: UP       Weight: 1
        Probes: 3       Failed [Total: 0 Current: 0]
        Last response: Success - ICMP echo reply received.
        Response Time: 1.275 millisec
 Done
```

1. In the details pane, click Save.
2. In the Save Config dialog box, click Yes.
3. Navigate to Traffic Management > Load Balancing > Virtual Servers.
4. In the details pane, select the virtual server that you created in step 5 and verify that the settings displayed in the Details pane are correct.
5. Navigate to Traffic Management > Load Balancing > Services.
6. In the details pane, select the service that you created in step 5 and verify that the settings displayed in the Details pane are correct.

After the configuration is up and running, you should view the statistics for each service and virtual server to check for possible problems.

Viewing the Statistics of a Virtual Server

Updated: 2013-09-05

To evaluate the performance of virtual servers or to troubleshoot problems, you can display details of the virtual servers configured on the NetScaler appliance. You can display a summary of statistics for all the virtual servers, or you can specify the name of a virtual server to display the statistics only for that virtual server. You can display the following details:

- Name
- IP address
- Port
- Protocol
- State of the virtual server
- Rate of requests received
- Rate of hits

To display a summary of the statistics for all the virtual servers currently configured on the NetScaler, or for a single virtual server, at the command prompt, type:

stat lb vserver [-detail] [<name>]

Example

```
>stat lb vserver -detail
Virtual Server(s) Summary
                vsvrIP  port    Protocol      State   Req/s   Hits/s
One                 *   80        HTTP         UP      5/s     0/s
```



```
Two                 *     0        TCP        DOWN      0/s      0/s
Three               *  2598        TCP        DOWN      0/s      0/s
dnsVirtualNS    10.102.29.90    53        DNS        DOWN      0/s      0/s
BRVSERV          10.10.1.1    80        HTTP        DOWN      0/s      0/s
LBVIP         10.102.29.66    80        HTTP          UP      0/s      0/s
 Done
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. If you want to display the statistics for only one virtual server, in the details pane, select the virtual server whose statistics you want to display.
3. In the details pane, click Statistics.

Viewing the Statistics of a Service

Updated: 2013-08-28

You can view the rate of requests, responses, request bytes, response bytes, current client connections, requests in surge queue, current server connections, and so forth using the service statistics.

At the command prompt, type:

stat service <name>
Example

```
stat service Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service whose statistics you want to view (for example, Service-HTTP-1).
3. Click Statistics. The statistics appear in a new window.

Global Server Load Balancing

Mar 22, 2012

NetScaler appliances configured for global server load balancing (GSLB) provide for disaster recovery and ensure continuous availability of applications by protecting against points of failure in a wide area network (WAN). GSLB can balance the load across data centers by directing client requests to the closest or best performing data center, or to surviving data centers in case of an outage.

Following are some typical GSLB configurations:

- **Active-active data center setup**. Consists of multiple active data centers. Client requests are load balanced across active data centers.
- **Active-standby data center setup**. Consists of an active and a standby data center. When a failover occurs as a result of a disaster event, the standby data center becomes operational.
- **Proximity setup**. Directs client requests to the data center that is closest in geographical distance or network distance.

In a typical configuration, a local DNS server sends client requests to a GSLB virtual server, to which are bound GSLB services. A GSLB service identifies a load balancing or content switching virtual server, which can be at the local site or a remote site. If the GSLB virtual server selects a load balancing or content switching virtual server at a remote site, it sends the virtual server's IP address to the DNS server, which sends it to the client. The client then resends the request to the new virtual server at the new IP.

The GSLB entities that you must configure are the GSLB sites, the GSLB services, the GSLB virtual servers, load balancing or content switching virtual servers, and authoritative DNS (ADNS) services. You must also configure MEP. You can also configure DNS views to expose different parts of your network to clients accessing the network from different locations.

Note: To take full advantage of the NetScaler GSLB features, you should use NetScaler appliances for load balancing or content switching at each data center, so that your GSLB configuration can use the proprietary Metric Exchange Protocol (MEP) to exchange site metrics.

How GSLB Works

May 20, 2015

With ordinary DNS, when a client sends a domain name system (DNS) request, it receives a list of IP addresses of the domain or service. Generally, the client chooses the first IP address in the list and initiates a connection with that server. The DNS server uses a technique called DNS round robin to rotate through the IPs on the list, sending the first IP address to the end of the list and promoting the others after it responds to each DNS request. This technique ensures equal distribution of the load, but it does not support disaster recovery, load balancing based on load or proximity of servers, or persistence.

When you configure GSLB on NetScaler appliances and enable Metric Exchange Protocol (MEP), the appliances use the DNS infrastructure to connect the client to the data center that best meets the criteria that you set. The criteria can designate the least loaded data center, the closest data center, the data center that responds most quickly to requests from the client's location, a combination of those metrics, and SNMP metrics. An appliance keeps track of the location, performance, load, and availability of each data center and uses these factors to select the data center to which to send a client request.

A GSLB configuration consists of a group of GSLB entities on each appliance in the configuration. These entities include GSLB sites, GSLB services, GSLB virtual servers, load balancing and/or content switching servers, and ADNS services.

This document includes the following information:

- GSLB Sites
- GSLB Services
- GSLB Virtual Servers
- Load Balancing or Content Switching Virtual Servers
- ADNS Services
- DNS VIPs

A typical GSLB setup consists of data centers, each of which has various network appliances that may or may not be NetScaler appliances. The data centers are called GSLB sites. Each GSLB site is managed by a NetScaler appliance that is local to that site. Each of these appliances treats its own site as the local site and all other sites, managed by other appliances, as remote sites.

If the appliance that manages a site is the only NetScaler appliance in that data center, the GSLB site hosted on that appliance acts as a bookkeeping placeholder for auditing purposes, because no metrics can be collected. Typically, this happens when the appliance is used only for GSLB, and other products in the data center are used for load balancing or content switching.

A GSLB service is usually a representation of a load balancing or content switching virtual server, although it can represent any type of virtual server. The GSLB service identifies the virtual server's IP address, port number, and service type. GSLB services are bound to GSLB virtual servers on the NetScaler appliances managing the GSLB sites. A GSLB service bound to a GSLB virtual server in the same data center is local to the GSLB virtual server. A GSLB service bound to a GSLB virtual server in

a different data center is remote from that GSLB virtual server.

A GSLB virtual server has one or more GSLB services bound to it, and load balances traffic among those services. It evaluates the configured GSLB methods (algorithms) to select the appropriate service to which to send a client request. Because the GSLB services can represent either local or remote servers, selecting the optimal GSLB service for a request has the effect of selecting the data center that should serve the client request.

The domain for which global server load balancing is configured must be bound to the GSLB virtual server, because one or more services bound to the virtual server will serve requests made for that domain.

Unlike other virtual servers configured on a NetScaler appliance, a GSLB virtual server does not have its own virtual IP address (VIP).

Updated: 2013-09-13

A load balancing or content switching virtual server represents one or many physical servers on the local network. Clients send their requests to the load balancing or content switching virtual server's virtual IP (VIP) address, and the virtual server balances the load across the physical servers. After a GSLB virtual server selects a GSLB service representing either a local or a remote load balancing or content switching virtual server, the client sends the request to that virtual server's VIP address.

For more information about load balancing or content switching virtual servers and services, see Load Balancing, or Content Switching .

An ADNS service is a special kind of service that responds only to DNS requests for domains for which the NetScaler appliance is authoritative. When an ADNS service is configured, the appliance owns that IP address and advertises it. Upon reception of a DNS request by an ADNS service, the appliance checks for a GSLB virtual server bound to that domain. If a GSLB virtual server is bound to the domain, it is queried for the best IP address to which to send the DNS response.

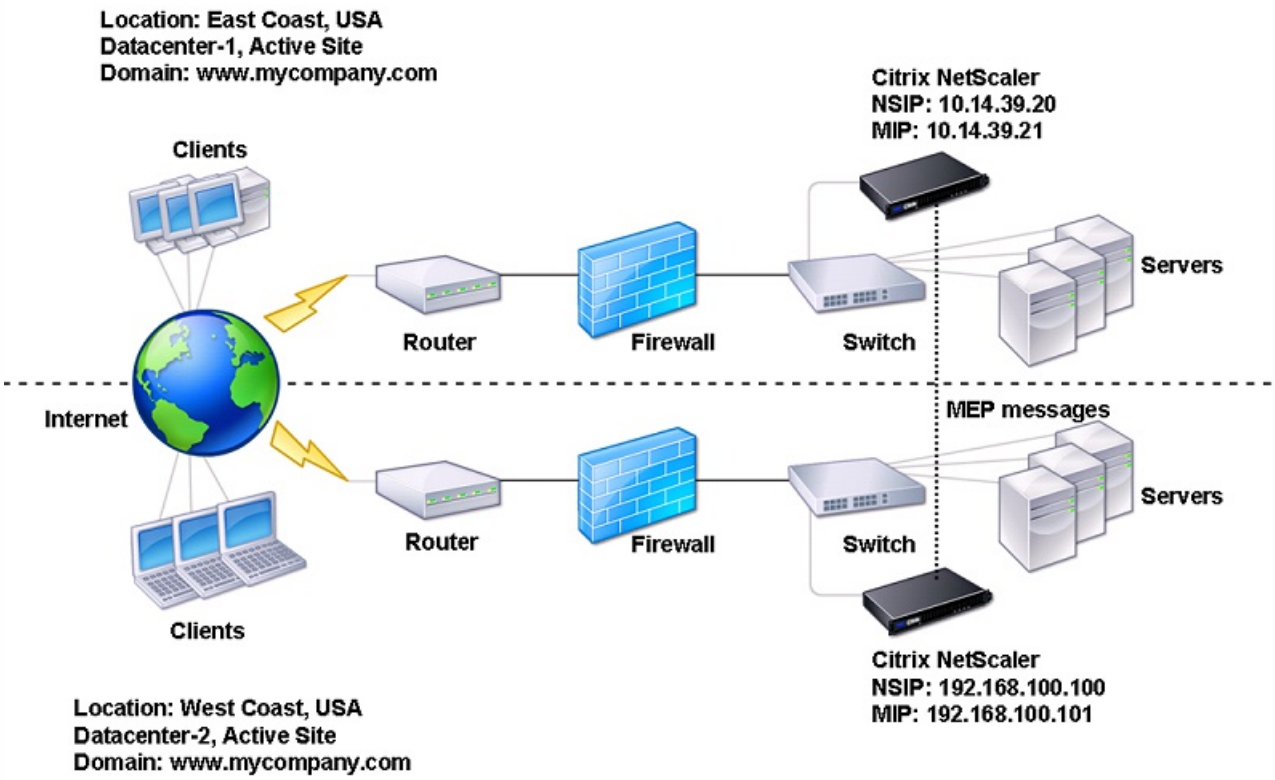
A DNS virtual IP is a virtual IP (VIP) address that represents a load balancing DNS virtual server on the NetScaler appliance. DNS requests for domains for which the NetScaler appliance is authoritative can be sent to a DNS VIP.

Configuring Global Server Load Balancing (GSLB)

May 22, 2015

Global server load balancing is used to manage traffic flow to a web site hosted on two separate server farms that ideally are in different geographic locations. For example, consider a Web site, www.mycompany.com, which is hosted on two geographically separated server farms or data centers. Both server farms use NetScaler appliances. The NetScaler appliances in these server farms are set up in one-arm mode and function as authoritative DNS servers for the www.mycompany.com domain. The following figure illustrates this configuration.

Figure 1. Basic GSLB Topology



To configure such a GSLB setup, you must first configure a standard load balancing setup for each server farm or data center. This enables you to balance load across the different servers in each server farm. Then, configure both NetScaler appliances as authoritative DNS (ADNS) servers. Next, create a GSLB site for each server farm, configure GSLB virtual servers for each site, create GLSB services, and bind the GSLB services to the GSLB virtual servers. Finally, bind the domain to the GSLB virtual servers. The GSLB configurations on the two appliances at the two different sites are identical, although the load-balancing configurations for each site is specific to that site.

Note: To configure a GSLB site in a NetScaler cluster setup, see Setting Up GSLB in a Cluster.

Updated: 2013-08-30

A load balancing virtual server balances the load across different physical servers in the data center. These servers are

represented as services on the NetScaler appliance, and the services are bound to the load balancing virtual server.

For details on configuring a basic load balancing setup, see Load Balancing.

Configuring an Authoritative DNS Service

Nov 24, 2014

When you configure the NetScaler appliance as an authoritative DNS server, it accepts DNS requests from the client and responds with the IP address of the data center to which the client should send requests.

Note: For the NetScaler to be authoritative, you must also create SOA and NS records. For more information about SOA and NS records, see "Domain Name System".

At the command prompt, type the following commands to create an ADNS service and verify the configuration:

- add service <name> <IP>@ ADNS <port>
- show service <name>

Example

```
add service Service-ADNS-1 10.14.39.21 ADNS 53
show service Service-ADNS-1
```

At the command prompt, type the following command:

set service <name> <IPAddress> ADNS <port>

Example

```
set service Service-ADNS-1 10.14.39.21 ADNS 53
```

At the command prompt, type the following command:

rm service <name>

Example

```
rm service Service-ADNS-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, do one of the following:
   - To create a new service, click Add.
   - To modify an existing service, select the service, and then click Open.
3. In the Create Service or Configure Service dialog box, specify values for the following parameters as shown:
   - Service Name*—name
   - Service Name*—name
   - Protocol*—(Select ADNS as the protocol.)
   - Port*—port
4. Click Create or OK, and then click Close. The server that you created appears in the GSLB Services pane.

Configuring a Basic GSLB Site

Nov 24, 2014

A GSLB site is a representation of a data center in your network and is a logical grouping of GSLB virtual servers, services, and other network entities. Typically, in a GSLB set up, there are many GSLB sites that are equipped to serve the same content to a client. These are usually geographically separated to ensure that the domain is active even if one site goes down completely. All of the sites in the GSLB configuration must be configured on every NetScaler appliance hosting a GSLB site. In other words, at each site, you configure the local GSLB site and each remote GSLB site.

Once GSLB sites are created for a domain, the NetScaler appliance sends client requests to the appropriate GSLB site as determined by the GSLB algorithms configured.

At the command prompt, type the following commands to create a GSLB site and verify the configuration:

- add gslb site <siteName> <siteIPAddress>
- show gslb site <siteName>

Example

```
add gslb site Site-GSLB-East-Coast 10.14.39.21
show glsb site Site-GSLB-East-Coast
```

- To modify a GSLB site, use the set gslb site command, which is just like using the add gslb site command, except that you enter the name of an existing GSLB Site.
- To unset a site parameter, use the unset gslb site command, followed by the siteName value and the name of the parameter to be reset to its default value.
- To remove a GSLB site, use the rm gslb site command, which accepts only the <name> argument.

1. Navigate to Traffic Management > GSLB > Sites.
2. In the details pane, do one of the following:
   - To create a new site, click Add.
   - To modify an existing site, select the site, and then click Open.
3. In the Create GSLB Site or Configure GSLB Site dialog box, specify values for the following parameters as shown:
   - Name*—siteName
   - Site IP Address*—siteIPAddress

   * A required parameter
4. Click Create or OK, and then click Close. The GSLB site you created appears in the GSLB Sites pane.

At the command prompt, type:

stat gslb site <siteName>

Example

```
stat gslb site Site-GSLB-East-Coast
```

1. Navigate to Traffic Management > GSLB > Sites.
2. In the GSLB Sites pane, select the GSLB site whose statistics you want to view.
3. Click Statistics.

Configuring a GSLB Service

Nov 24, 2014

A GSLB service is a representation of a load balancing or content switching virtual server. A local GSLB service represents a local load balancing or content switching virtual server. A remote GSLB service represents a load balancing or content switching virtual server configured at one of the other sites in the GSLB setup. At each site in the GSLB setup, you can create one local GSLB service and any number of remote GSLB services.

To create a GSLB service by using the command line interface

At the command prompt, type the following commands to create a GSLB service and verify the configuration:

- add gslb service <serviceName> <serverName | IP> <serviceType> <port>-siteName <string>
- show gslb service <serviceName>

Example

```
add gslb service Service-GSLB-1 10.14.39.14 HTTP 80 –siteName Site-GSLB-East-Coast
show gslb service Service-GSLB-1
```

To modify or remove a GSLB service by using the command line interface

- To modify a GSLB service, use the set gslb service <serviceName> command. For this command, specify the name of the GSLB service whose configuration you want to modify. You can change the existing values of the parameters either specified by you or set by default. You can change the value of more than one parameter in the same command. Refer to the add gslb service command for details about the parameters. Example
  ```
  > set gslb service SKP_GSLB_NOTCNAME_SVC2 -maxBandWidth 25 -maxClient 8
  Done
  > sh gslb service SKP_GSLB_NOTCNAME_SVC2
  SKP_GSLB_NOTCNAME_SVC2 (21.211.21.21: 80)- HTTP
  ...
  Max Conn: 8  Max Bandwidth: 25 kbits
  ```
- To reset a parameter to its default value, you can use the unset gslb service <serviceName> command and the parameters to be unset. Example
  ```
  > unset gslb service SKP_GSLB_NOTCNAME_SVC2 maxBandWidth
  Done
  > sh gslb service SKP_GSLB_NOTCNAME_SVC2
  SKP_GSLB_NOTCNAME_SVC2 (21.211.21.21: 80)- HTTP
  ...
  Max Conn: 8  Max Bandwidth: 0 kbits
  ```
- To remove a GSLB service, use the rm gslb service <serviceName> command.

To create a GSLB service by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. In the details pane, do one of the following:
   - To create a new service, click Add.

- To modify an existing service, select the service, and then click Open.

3. In the Create GSLB Service or Configure GSLB Service dialog box, set the following parameters:
   - Service Name*
   - Site Name*
   - Server Name - The servers added to the NetScaler configuration are displayed in a dropdown list. If you want to add a new server, click New..., and then in the Create Server dialog box, type the necessary details. For more information about creating servers, see "Adding a Server."
   - Service Type
   - Port

   Note: In the Site Name and Server Name lists, the most recently used value is displayed as selected. Make sure that you select the site and server you want to specify.
4. Click Create, and then click Close. The GSLB service you created appears in the GSLB Services pane.

To view the statistics of a GSLB service by using the command line interface

At the command prompt, type:

stat gslb service <serviceName>

Example

```
stat gslb service Service-GSLB-1
```

To view the statistics of a GSLB service by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Services pane, select the GSLB Service whose statistics you want to view.
3. Click Statistics.

Updated: 2013-08-30

Before you use a GSLB service for load balancing, it must be enabled. If the service is disabled, it is not included in load balancing even though it exists on the NetScaler appliance.

To enable or disable a GSLB service by using the command line interface

At the command prompt, type one of the following commands:

- enable service <name>
- disable service <name>

Example

```
> enable service Service-GSLB-1
 Done
> disable service Service-GSLB-1
 Done
```

To enable or disable a GSLB service by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.

2. In the GSLB Services pane, select the GSLB service which you want to enable or disable.
3. Click enable or disable.

Configuring a GSLB Virtual Server

Dec 09, 2014

A GSLB virtual server is an entity that represents one or more GSLB services and balances traffic between them. It evaluates the configured GSLB methods or algorithms to select a GSLB service to which to send the client request.

To create a GSLB virtual server by using the command line interface

At the command prompt, type the following commands to add a GSLB virtual server and verify the configuration:

- add gslb vserver <name> <serviceType> -ipType (IPv4 | IPv6)
- show gslb vserver <name>

Example

```
add gslb vserver Vserver-GSLB-1 HTTP -ipType IPv4
add gslb vserver Vserver-GSLB-2 HTTP -ipType IPv6
show gslb vserver Vserver-GSLB-1
show gslb vserver Vserver-GSLB-2
```

To modify or remove a GSLB virtual server by using the command line interface

- To modify a GSLB virtual server, use the set gslb vserver command, which is just like using the add gslb vserver command, except that you enter the name of an existing GSLB virtual server.
- To reset a parameter to is default value, you can use the unset gslb vserver command followed by the vserverName value and the name of the parameter to be unset.
- To remove a GSLB virtual server, use the rm gslb vserver command, which accepts only the <name> argument.

To configure a GSLB virtual server by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. Add a new GSLB virtual server, or select an existing GSLB virtual server and edit its settings.

To view the statistics of a GSLB virtual server by using the command line interface

At the command prompt, type:

stat gslb vserver <name>

Example

```
stat gslb vserver Vserver-GSLB-1
```

To view the statistics of a GSLB virtual server by using the configuration utility

Navigate to Traffic Management > GSLB > Virtual Servers, select the virtual server and click **Statistics**.

Statistics of a GSLB service

When you run the stat gslb service command from the command line or click on the Statistics link from the configuration utility, the following details of the service will be displayed:

- **Request bytes.** Total number of request bytes received on this service or virtual server.
- **Response bytes.** Number of response bytes received by this service or virtual server.
- **Current client established connections.** Number of client connections in ESTABLISHED state.
- **Current load on the service.** Load on the service (Calculated from the load monitor bound to the service).

The data of number of requests and responses, and the number of current client and server connections may not be displayed or may not be synchronized with the data of the corresponding load balancing virtual server.

Updated: 2014-11-21

When you create a GSLB virtual server, it is enabled by default. If you disable it, it cannot process traffic. A disabled GSLB virtual server is not included in GSLB configuration but is not removed from the NetScaler appliance.

To enable or disable a GSLB virtual server by using the command line interface

At the command prompt, type one of the following commands:

- enable gslb vserver <name>@
- disable gslb vserver <name>@

Example

```
enable gslb vserver Vserver-GSLB-1
disable gslb vserver Vserver-GSLB-1
```

To enable or disable a GSLB virtual server by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. Select a virtual server and, from the **Action** list, select **enable** or **disable**.

Binding GSLB Services to a GSLB Virtual Server

Nov 21, 2014

Once the GSLB services and virtual server are configured, relevant GSLB services must be bound to the GSLB virtual server to activate the configuration.

At the command prompt, type the following commands to bind a GSLB service to a GSLB virtual server and verify the configuration:

- bind gslb vserver <name> -serviceName <string>
- show gslb vserver <name>

Example

```
bind gslb vserver Vserver-GSLB-1 -serviceName Service-GSLB-1
show gslb vserver Vserver-GSLB-1
```

At the command prompt, type:

unbind gslb vserver <name> -serviceName <string>

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the details pane, select the GSLB Virtual Server to which you want to bind the services (for example, Vserver-GSLB-1).
3. Click Open.
4. On the Services tab, in the Active column, select the check boxes next to the GSLB services that you want to bind to the GSLB virtual server.
5. Click OK.

Binding a Domain to a GSLB Virtual Server

Dec 09, 2014

To make a NetScaler appliance the authoritative DNS server for a domain, you must bind the domain to the GSLB virtual server. When you bind a domain to a GSLB virtual server, the NetScaler adds an address record for the domain, containing the name of the GSLB virtual server. The start of authority (SOA) and name server (NS) records for the GSLB domain must be added manually.

For details on configuring SOA and NS records, see "Domain Name System".

At the command prompt, type the following commands to bind a domain to a GSLB virtual server and verify the configuration:

- bind gslb vserver <name> -domainName <string>
- show gslb vserver <name>

Example

```
bind gslb vserver Vserver-GSLB-1 -domainName www.mycompany.com
show gslb vserver Vserver-GSLB-1
```

At the command prompt, type:

unbind gslb vserver <name> -domainName <string>

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In GSLB Virtual Servers pane, select the GSLB Virtual Server to which you want to bind the domain (for example, Vserver-GSLB-1) and click Open.
3. In the Configure GSLB Virtual Server dialog box, on the Domains tab, do one of the following:
   - To create a new Domain, click Add.
   - To modify an existing Domain, select the domain, and then click Open.
4. In the Create GSLB Domain or Configure GSLB Domain dialog box, specify values for the following parameters as shown:
   - Domain Name*—domainName (for example, www.mycompany.com)

   * A required parameter
5. Click Create.
6. Click OK.

At the command prompt, type:

stat gslb domain <name>

Example

```
stat gslb domain www.mycompany.com
```

Note: To view statistics for a particular GSLB domain, enter the name of the domain exactly as it was added to the

NetScaler appliance. If you do not specify the domain name, or if you specify an incorrect domain name, statistics for all configured GSLB domains are displayed.

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In GSLB Virtual Servers pane, select the GSLB Virtual Server (for example, Vserver-GSLB-1) and click Open.
3. In the Configure GSLB Virtual Server dialog box, on the Domains tab, select the domain, and then click Statistics.

Synchronizing a Configuration in a GSLB Setup

Mar 24, 2015

Typically, a GSLB setup has a few data centers with a GSLB site configured for each data center. In each NetScaler, participating in GSLB, configure one GSLB site as a local site and the others as remote sites. When you add another GSLB site at a later point of time, ensure that all the GSLB sites have the same configuration. To have the same configuration on all the GSLB sites, you can use the NetScaler appliance's GSLB configuration synchronization option.

The NetScaler appliance from which you use the synchronization option is referred to as the 'master node' and the GSLB sites on which the configuration is copied as 'slave nodes'. When you synchronize a GSLB configuration, the configurations on all the GSLB sites participating in the GSLB setup are made similar to that on the master node.

Synchronization (may also be referred to as 'auto sync') is carried out in the following manner:

- The master node finds the differences between the configuration of the master node and slave node, and changes the configuration of the slave node to make it similar to the master node.
  If you force a synchronization (use the 'force sync' option), the NetScaler deletes the GSLB configuration from the slave node and then configures the slave to make it similar to the master node.
- During synchronization, if a command fails, synchronization is not aborted.
- Synchronization is done only on the parent sites. If a GSLB site is configured as a child site, its configuration is not affected by synchronization.

Note: On the remote GSLB site RPC node, configure the firewall to accept auto-sync connections by specifying the remote site IP (cluster IP address for cluster setup) and port (3010 for RPC and 3008 for secure RPC). The source IP address that will be used for auto-sync is the NSIP of the master node (NSIP of the configuration coordinator in a cluster setup).
If you use the saveconfig option, the sites that participate in the synchronization process automatically save their configuration, in the following way:

1. The master node saves its configuration immediately before it initiates the process of synchronization.
2. After the process of synchronization is complete, the slave nodes save their configuration. A slave node saves its configuration only if the configuration difference was applied successfully on it. If synchronization fails on a slave node, you must manually investigate the cause of the failure and take corrective action.

Limitations of synchronization:

- On the master node, the names of the remote GSLB sites must be identical to the names of sites configured on the NetScaler appliances hosting those sites.
- During the synchronization, traffic disruptions may occur.
- NetScaler can synchronize only up to 80000 lines of the configuration.
- Synchronization may fail:
  - If the spill over method is changed from CONNECTION to DYNAMIC CONNECTION.
  - If you interchange the site prefix of the GSLB services bound to a GSLB virtual server on the master node and then try to synchronize.
  - If the RPC node passwords are different for NetScaler IP address (NSIP) and loopback IP address.
- If you have configured the GSLB sites as High Availability (HA) pairs, the RPC node passwords of primary and secondary nodes should be same.
- If you rename any GLSB entity that are part of your GSLB configuration (use “show gslb runningConfig” command to display the GSLB configuration). You need to use the force sync option to synchronize the configuration to other GSLB sites.

Note: To overcome the limitations due to some settings in the GSLB configuration, you can use the force sync option. But, if you use the force sync option the GSLB entities are removed and re-added to the configuration and the GSLB statistics are reset to zero. Hence the traffic is disrupted during the configuration change.
Before you start the synchronization of a GSLB setup, make sure that:

- On all the GSLB sites including the master node, management access should be enabled for the IP address of the corresponding GSLB site. The IP address of a GSLB site must be an IP address owned by the NetScaler.
  For more information about adding the GSLB site IP addresses and enabling Management Access, see "Configuring a Basic GSLB Site" and "Configuring NetScaler-Owned IP Addresses".
- The GSLB configuration on the NetScaler appliance that is considered as the master node is complete and appropriate to be copied on all the sites.
- If you are synchronizing the GSLB configuration for the first time, all the sites participating in GSLB need to have the GSLB site entity of their respective local sites.
- You are not synchronizing sites that, by design, do not have the same configuration.

Important: After a GSLB configuration is synchronized, the configuration cannot be rolled back on any of the GSLB sites. Run the **sync gslb config** command only if you are sure that the synchronization process will not overwrite the configuration on the remote site. Site synchronization is undesirable when the local and remote sites have different configurations by design, and can lead to site outage. If some commands fail and some commands succeed, the successful commands cannot be rolled back.

At the command prompt, type the following commands to synchronize GSLB sites and verify the configuration:

- sync gslb config [-preview | -forceSync <string> | -nowarn | -saveconfig] [-debug]
- show gslb syncStatus

Example

```
> sync gslb config
[WARNING]: Syncing config may cause configuration loss on other site.
Please confirm whether you want to sync-config (Y/N)? [N]:y
Sync Time: Dec 9 2011 10:56:9
Retrieving local site info: ok
Retrieving all participating gslb sites info: ok
Gslb_site1[Master]:
      Getting Config: ok
Gslb_site2[Slave]:
      Getting Config: ok
      Comparing config: ok
       Applying changes: ok
Done
```

1. Navigate to Traffic Management > GSLB.
2. In the GSLB pane, under GSLB Configuration, click Synchronize configuration on remote sites.
3. In the Synchronize GSLB Configuration dialog box, select one of the following settings from the Synchronization Option list:

- Preview
- Force Sync
- Debug

4. If you select Force Sync as the synchronization option, in the GSLB Site Name text box, type the name of the remote site that you want to synchronize with the local site, or type all-sites.
5. If you want the participating sites to save their configuration automatically, select Save Configuration.
6. Click Run.
7. If you want to save the output of the Run command to your local system, click Save output text to a file.
8. Click Close.

Viewing and Configuring a GSLB Setup by Using the GSLB Visualizer

Sep 23, 2013

The configuration utility includes a GSLB Visualizer tool, which provides an alternative way to view and configure entities in a GSLB configuration. The visualizer displays all configured GSLB domains, GSLB services, GSLB sites, ADNS services, and any monitors that are bound to the services. It also displays all the load balancing, content switching, cache redirection, and NetScaler Gateway virtual servers that the GSLB services represent.

If you want to view the configurations of remote GSLB sites, you must configure the sites with public IP addresses and enable management access for each of them.

You can use the GSLB Visualizer to perform the following GSLB configuration tasks:

- Add, view, and configure GSLB domains and GSLB services.
- View and configure GSLB sites and ADNS services for each site.
- View and configure any monitors that are bound to the services.
- View and configure the content switching, load balancing, cache redirection, or NetScaler Gateway virtual server that each GSLB service represents.
- View statistics for GSLB domains, sites, ADNS services, and virtual servers.
- View configuration details of any displayed entity.
- View load balancing and content switching virtual servers.
- View bindings for GSLB services, ADNS services, monitors, and virtual servers.
- Enable and disable GSLB services, ADNS services, monitors, and virtual servers.
- Copy the properties of any displayed entity to a document or spreadsheet.
- Remove a domain from the GSLB setup.
- Save the visual representation of the GSLB setup as an image.

1. Navigate to Traffic Management > GSLB.
2. In the details pane, under Getting Started, click GSLB Visualizer, and then do the following.
   - To pan the view of the displayed image, click as blank area of the image, hold down the mouse button, and drag the image.
   - To adjust the viewable area click Zoom In to increase or Zoom Out to decrease the size of the objects. You can readjust the viewable area by clicking Best Fit.
   - To locate a specific item, begin typing the item's name in the Search field. Entities whose names match the typed characters are highlighted. Continue typing until the item is uniquely identified. To clear the Search field, click the x adjacent to the field.

1. Open the GSLB Visualizer and click Domain. Alternatively, if domains already exist in the GSLB setup, click the name of an existing domain.
2. Under Related Tasks, click Add.
3. Follow the instructions in the GSLB Wizard to add a GSLB domain and configure GSLB services and sites for the domain.

Open the GSLB Visualizer and do one of the following:

- To view a brief summary of an entity, place the pointer on the entity. A brief summary of the entity appears at the bottom of the viewable area.
- To view the detailed configuration information of the entity, click the entity. The configuration details for that entity appear in the Details area.

Open the GSLB Visualizer and do one of the following:

- Click the entity that you want to modify. Then, under Related Tasks, click Open.
- Double-click the entity that you want to modify.
- Right-click the entity that you want to modify, and then click Open. (This option is not available for GSLB sites.)

Open the GSLB Visualizer and do one of the following:

- Click the entity whose binding information you want to view, and then, under Related Tasks, click Show Bindings.
- Right-click the entity, and then click Show Bindings.

Open the GSLB Visualizer and do one of the following:

- Click the load balancing or content switching virtual server whose Visualizer you want to view, and then, under Related Tasks, click Visualizer.
- Right-click the virtual server, and then click Visualizer.

Open the GSLB Visualizer and do one of the following:

- Click the entity whose statistics you want to view, and then, under Related Tasks, click Statistics.
- Right-click the entity whose statistics you want to view, and then click Statistics. (This option is not available for GSLB sites.)

Open the GSLB Visualizer and do one of the following to enable or disable the entity:

- To enable the entity, click the entity and, under Related Tasks, click Enable. Alternatively, right-click the entity that you want to enable, and then click Enable.
- To disable the entity, click the entity and, under Related Tasks, click Disable. Alternatively, right-click the entity that you want to disable, and then click Disable.

Open the GSLB Visualizer and do one of the following:

- Click the entity whose properties you want to copy, and then, under Related Tasks, click Copy Properties.
- Right-click the entity, and then click Copy. (This option is not available for GSLB sites.)

1. Open the GSLB Visualizer.

2. If necessary, adjust the viewable area by using the Best Fit, Zoom In, and Zoom Out buttons.
3. Click Save Image.
4. In the Save Graph Image dialog box, browse to the folder in which you want to save the image.
5. In File Name text box, type the name, and then click Save.

1. Open the GSLB Visualizer and do one of the following:
   - Click the domain that you want to remove, and then, under Related Tasks, click Remove.
   - Right-click the domain, and then click Remove.
2. Under Remove?, click Yes.

Configuring the Metrics Exchange Protocol (MEP)

May 21, 2015

The data centers in a GSLB setup exchange metrics with each other through the metrics exchange protocol (MEP), which is a proprietary protocol for the Citrix NetScaler. The exchange of the metric information begins when you create a GSLB site. These metrics comprise load, network, and persistence information.

MEP is required for health checking of data centers to ensure their availability. A connection for exchanging network metrics can be initiated by either of the data centers involved in the exchange, but a connection for exchanging site metrics is always initiated by the data center with the lower IP address. By default, the data center uses a subnet IP address (SNIP) or a mapped IP address (MIP) to establish a connection to the IP address of a different data center. However, you can configure a specific SNIP, MIP, the NetScaler IP address (NSIP), or a virtual IP address (VIP) as the source IP address for metrics exchange. The communication process between GSLB sites uses TCP port 3011 or 3009, so this port must be open on firewalls that are between the NetScaler appliances.
Note: You cannot configure a GSLB site IP address as the source IP address for site metrics exchange.
If the source and target sites for a MEP connection (the site that initiates a MEP connection and the site that receives the connection request, respectively) have both private and public IP addresses configured, the sites exchange MEP information by using the public IP addresses.

You can also bind monitors to check the health of remote services. When monitors are bound, metric exchange does not control the state of the remote service. If a monitor is bound to a remote service and metrics exchange is enabled, the monitor controls the health status. Binding the monitors to the remote service allows the NetScaler to interact with a non-NetScaler load balancing device. The NetScaler can monitor non-NetScaler devices but cannot perform load balancing on them. The NetScaler can monitor non-NetScaler devices, and can perform load balancing on them if monitors are bound to all GSLB services and only static load balancing methods (such as the round robin, static proximity, or hash-based methods) are used.

This document includes the following information:

- Configuring Site Metric Exchange
- Configuring Network Metric Information Exchange
- Configuring Persistence Information Exchange

Updated: 2014-11-24

Site metrics exchanged between the GSLB sites include the status of each load balancing and content switching virtual server, the current number of connections, the current packet rate, and current bandwidth usage information.

The NetScaler appliance needs this information to perform load balancing between the sites. The site metric exchange interval is 1 second. A remote GSLB service must be bound to a local GSLB virtual server to enable the exchange of site metrics with the remote service.

To enable or disable site metric exchange by using the command line interface

At a command prompt, type the following commands to enable or disable site metric exchange and verify the configuration:

- set gslb site <siteName> -metricExchange(ENABLED | DISABLED)
- show gslb site <siteName>

Example

```
set gslb site Site-GSLB-East-Coast -metricExchange ENABLED
set gslb site Site-GSLB-East-Coast -metricExchange DISABLED
show gslb site Site-GSLB-East-Coast
```

To enable or disable site metric exchange by using the configuration utility

1. Navigate to Traffic Management > GSLB > Sites.
2. In the details pane, select the site, and then click Open.
3. In the Configure GSLB Site dialog box, select or clear the check box next to the Metric Exchange and click OK.

Updated: 2014-11-24

You can enable or disable the exchange of round trip time (RTT) information about the client's local DNS when the GSLB dynamic method (RTT) is enabled. This information is exchanged every 5 seconds.

For details about changing the GSLB method to a method based on RTT, see Changing the GSLB Method.

To enable or disable network metric information exchange by using the command line interface

At the command prompt, type the following commands to enable or disable network metric information exchange and verify the configuration:

- set gslb site <siteName> -nwmetricExchange (ENABLED | DISABLED)
- show gslb site <<siteName>

Example

```
set gslb site Site-GSLB-East-Coast -nwmetricExchange ENABLED
set gslb site Site-GSLB-East-Coast -nwmetricExchange DISABLED
show gslb site Site-GSLB-East-Coast
```

To enable or disable network metric information exchange by using the configuration utility

1. Navigate to Traffic Management > GSLB > Sites.
2. In the details pane, select the site, and then click Open.
3. In the Configure GSLB Site dialog box, select or clear the check box next to the Network Metric Exchange and click OK.

Updated: 2014-11-24

You can enable or disable the exchange of persistence information at each site. This information is exchanged every 5

seconds between NetScaler appliances participating in GSLB.

For details about configuring persistence, see "Configuring Persistent Connections".

To enable/disable persistence information exchange by using the command line interface

At the command prompt, type the following commands to enable or disable persistence information exchange and verify the configuration:

- set gslb site <siteName> -sessionExchange (ENABLED | DISABLED)
- show gslb site <siteName>

Example

```
set gslb site Site-GSLB-East-Coast -sessionExchange  ENABLED
set gslb site Site-GSLB-East-Coast -sessionExchange  DISABLED
show gslb site Site-GSLB-East-Coast
```

To enable/disable persistence information exchange by using the configuration utility

1. Navigate to Traffic Management > GSLB > Sites.
2. In the details pane, select the site, and then click Open.
3. In the Configure GSLB Site dialog box, select or clear the check box next to the Persistence Session Entry Exchange and click OK.

Configuring Site-to-Site Communication

Jun 02, 2015

GSLB site-to-site communication is between the remote procedure call (RPC) nodes that are associated with the communicating sites. A master GSLB site establishes connections with slave sites to synchronize GSLB configuration information and to exchange site metrics.

An RPC node is created automatically when a GSLB site is created, and is assigned an internally generated user name and password. The NetScaler appliance uses this user name and password to authenticate itself to remote GSLB sites during connection establishment. No configuration steps are necessary for an RPC node, but you can specify a password of your choice, enhance security by encrypting the information that GSLB sites exchange, and specify a source IP address for the RPC node.

The appliance needs a NetScaler-owned IP address to use as the source IP address when communicating with other GSLB sites. By default, the RPC nodes use either a subnet IP (SNIP) address or a mapped IP (MIP) address, but you might want to specify an IP address of your choice.

The following topics describe the behavior and configuration of RPC nodes on the NetScaler appliance:

- Changing the Password of an RPC Node
- Encrypting the Exchange of Site Metrics
- Configuring the Source IP Address for an RPC Node

Updated: 2014-11-21

You can secure the communication between sites in your GSLB setup by changing the password of each RPC node. After you change the password for the RPC node of the local site, you must manually propagate the change to the RPC node at each of the remote sites.

The password is stored in encrypted form. You can verify that the password has changed by using the **show rpcNode** command to compare the encrypted form of the password before and after the change.

To change the password of an RPC node by using the command line interface

At the command line, type the following commands to change the password of an RPC node:

- set ns rpcNode <IPAddress> {-password}
- show ns rpcNode

Example

```
> set rpcNode 192.0.2.4 -password mypassword
 Done
> show rpcNode
.
.
.
2) IPAddress:  192.0.2.4 Password:  d336004164d4352ce39e
 SrcIP:  *           Secure:  OFF
```

```
Done
>
```

To unset the password of an RPC node by using the command line interface

To unset the password of an RPC node by using the NetScaler command line, type the unset rpcNode command, the IP address of the RPC node, and the password parameter, without a value.

To change the password of an RPC node by using the configuration utility

1. Navigate to System > Network > RPC.
2. In the details pane, click the RPC node for which you want to change the password, and then click Open.
3. In the Configure RPC Node dialog box, in Password and Confirm Password, specify the password that you want the RPC node to use.

Updated: 2014-11-24

You can secure the information that is exchanged between GSLB sites by setting the secure option for the RPC nodes in the GSLB setup. With the secure option set, the NetScaler appliance encrypts all communication sent from the node to other RPC nodes.

To encrypt the exchange of site metrics by using the command line interface

At the command prompt, type the following commands to encrypt the exchange of site metrics and verify the configuration:

- set ns rpcNode <IPAddress> [-secure (YES | NO)]
- show rpcNode

Example

```
> set rpcNode 192.0.2.4 -secure YES
 Done
>
> show rpcNode
.
.
.
3) IPAddress: 192.0.2.4 Password: d336004164d4352ce39e SrcIP: 192.0.2.3    Secure: ON
 Done
>
```

To unset the secure parameter by using the command line interface

To unset the secure parameter by using the NetScaler command line, type the unset rpcNode command, the IP address of the RPC node, and the secure parameter, without a value.

To encrypt the exchange of site metrics by using the NetScaler configuration utility

1. Navigate to System > Network > RPC.
2. In the details pane, click the RPC node whose communication you want to encrypt, and then click Open.

3. In the Configure RPC Node dialog box, click Secure.
4. Click OK.

Updated: 2014-11-24

By default, the NetScaler appliance uses a NetScaler-owned subnet IP (SNIP) address or mapped IP (MIP) address as the source IP address for an RPC node, but you can configure the appliance to use a specific SNIP address or MIP address. If neither a SNIP address nor a MIP address is available, the GSLB site cannot communicate with other sites. In such a scenario, you must configure either the NetScaler IP (NSIP) address or a virtual IP (VIP) address as the source IP address for an RPC node. A VIP address can be used as the source IP address of an RPC node only if the RPC node is a remote node. If you configure a VIP address as the source IP address and remove the VIP address, the appliance uses a SNIP address or a MIP address.

To specify a source IP address for an RPC node by using the command line interface

At the command prompt, type the following commands to change the source IP address for an RPC node and verify the configuration:

- set ns rpcNode <IPAddress> [-srcIP <ip_addr|ipv6_addr|*>]
- show ns rpcNode

Example

```
> set rpcNode 192.0.2.4 -srcIP 192.0.2.3
 Done
> show rpcNode
.
.
.
2) IPAddress:  192.0.2.4 Password:  d336004164d4352ce39e SrcIP:  192.0.2.3      Secure:  OFF
 Done
>
```

To unset the source IP address parameter by using the command line interface

To unset the source IP address parameter by using the NetScaler command line, type the unset rpcNode command, the IP address of the RPC node, and the srcIP parameter, without a value.

To specify a source IP address for an RPC node by using the NetScaler configuration utility

1. Navigate to System > Network > RPC.
2. In the details pane, click the RPC node for which you want to assign a specific source IP address for site metrics exchange, and then click Open.
3. In the Configure RPC Node dialog box, in Source IP Address, enter the IP address that you want the RPC node to use as the source IP address.

Customizing Your GSLB Configuration

May 21, 2015

Once your basic GSLB configuration is operational, you can customize it by modifying the bandwidth of a GSLB service, configuring CNAME based GSLB services, static proximity, dynamic RTT, persistent connections, or dynamic weights for services, or changing the GSLB Method.

You can also configure monitoring for GSLB services to determine their states.

These settings depend on your network deployment and the types of clients you expect to connect to your servers.

This document includes the following information:

- Modifying Maximum Connections or Maximum Bandwidth for a GSLB Service
- Creating CNAME-Based GSLB Services
- Configuring Transition Out-Of-Service State (TROFS) in GSLB
- Configuring Dynamic Weights for Services

Updated: 2014-11-26

You can restrict the number of new clients that can simultaneously connect to a load balancing or content switching virtual server by configuring the maximum number of clients and/or the maximum bandwidth for the GSLB service that represents the virtual server.

To modify the maximum clients or bandwidth of a GSLB service by using the command line interface

At the command prompt, type the following command to modify the maximum number of client connections or the maximum bandwidth of a GSLB service and verify the configuration:

- set gslb service <serviceName> [-maxClients <positive_integer>] [-maxBandwidth <positive_integer>]
- show gslb service <serviceName>

Example

```
set glsb service Service-GSLB-1 –maxBandwidth 100 –maxClients 100
show gslb service Service-GSLB-1
```

To modify the maximum clients or bandwidth of a GSLB service by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. In the details pane, select the service to be modified and click Open.
3. In the Configure GSLB Service dialog box specify values for one or both of the following parameters:
   - Max Clients—maxClients
   - Max Bandwidth—maxBandwidth

4. Click OK.
5. Verify that the Details area displays the values that you entered.

Updated: 2014-11-24

To configure a GSLB service, you can use the IP address of the server or a canonical name of the server. If you want to run multiple services (like an FTP and a Web server, each running on different ports) from a single IP address or run multiple HTTP services on the same port, with different names, on the same physical host, you can use canonical names (CNAMES) for the services.

For example, you can have two entries in DNS as ftp.example.com and www.example.com for FTP services and HTTP services on the same domain, example.com. CNAME-based GSLB services are useful in a multilevel domain resolver configuration or in multilevel domain load balancing. Configuring a CNAME-based GSLB service can also help if the IP address of the physical server is likely to change.

If you configure CNAME-based GSLB services for a GSLB domain, when a query is sent for the GSLB domain, the NetScaler appliance provides a CNAME instead of an IP address. If the A record for this CNAME record is not configured, the client must query the CNAME domain for the IP address. If the A record for this CNAME record is configured, the NetScaler provides the CNAME with the corresponding A record (IP address). The NetScaler appliance handles the final resolution of the DNS query, as determined by the GSLB method. The CNAME records can be maintained on a different NetScaler appliance or on a third-party system.

In an IP-address-based GSLB service, the state of a service is determined by the state of the server that it represents. However, a CNAME-based GSLB service has its state set to UP by default; the virtual server IP (VIP) address or metric exchange protocol (MEP) are not used for determining its state. If a desktop-based monitor is bound to a CNAME-based GSLB service, the state of the service is determined according to the result of the monitor probes.

You can bind a CNAME-based GSLB service only to a GSLB virtual server that has the DNS Record Type as CNAME. Also, a NetScaler appliance can contain at most one GSLB service with a given CNAME entry.

The following are some of the features supported for a CNAME-based GSLB service:
- GSLB-policy based site affinity is supported, with the CNAME as the preferred location.
- Source IP persistence is supported. The persistency entry contains the CNAME information instead of the IP address and port of the selected service.

The following are the limitations of CNAME-based GSLB services:
- Site persistence is not supported, because the service referenced by a CNAME can be present at any third-party location.
- Multiple-IP-address response is not supported because one domain cannot have multiple CNAME entries.
- Source IP Hash and Round Robin are the only load balancing methods supported. The Static Proximity method is not supported because a CNAME is not associated with an IP address and static proximity can be maintained only according to the IP addresses.

Note: The Empty-Down-Response feature should be enabled on the GSLB virtual server to which you bind the CNAME-based GSLB service. If you enable the Empty-Down-Response feature, when a GSLB virtual server is DOWN or disabled, the response to a DNS query, for the domains bound to this virtual server, contains an empty record without any IP addresses, instead of an error code.

To create a CNAME-based GSLB service by using the command line interface

At the command prompt, type:

add gslb service <serviceName> -cnameEntry <string> -siteName <string>

Example

```
add gslb service Service-GSLB-1 -cnameEntry transport.mycompany.com -siteName Site-GSLB-East-Coast
add gslb service Service-GSLB-2 -cnameEntry finance.mycompany.com -siteName Site-GSLB-West-Coast
```

To create a CNAME-based GSLB service by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. In the details pane, click Add.
3. In the Create GSLB Service dialog box, set the following parameters:
   - Service Name*
   - Site Name*
   - Type should be Canonical name based.
   - DNS Canonical name*

   * A required parameter
4. Click Create, and then click Close.

When you configure persistence on a GSLB virtual server to which a service is bound, the service continues to serve requests from the client even after it is disabled, accepting new requests or connections only to honor persistence. After a configured period of time, known as the graceful shutdown period, no new requests or connections are directed to the service, and all of the existing connections are closed.

When disabling a service, you can specify a graceful shutdown period, in seconds, by using the delay argument. During the graceful shutdown period, if the service is bound to a virtual server, its state appears as Out of Service.

Updated: 2015-06-02

In a typical network, there are servers that have a higher capacity for traffic than others. However, with a regular load balancing configuration, the load is evenly distributed across all services even though different services represent servers with different capacities.

To optimize your GSLB resources, you can configure dynamic weights on a GSLB virtual server. The dynamic weights can be based on either the total number of services bound to the virtual server or the sum of the weights of the individual services bound to the virtual server. Traffic distribution is then based on the weights configured for the services.

When dynamic weights are configured on the GSLB virtual server, requests are distributed according to the load balancing method, the weight of the GSLB service, and the dynamic weight. The product of the weight of the GSLB service and the dynamic weight is known as the cumulative weight. Therefore, when dynamic weight is configured on the GSLB virtual server, requests are distributed on the basis of the load balancing method and the cumulative weight.

When dynamic weight for a virtual server is disabled, the numerical value is set to 1. This ensures that the cumulative weight is a non-zero integer at all times.

Dynamic weight can be based on the total number of active services bound to load balancing virtual servers or on the weights assigned to the services.

Consider a configuration with two GSLB sites configured for a domain and each site has two services that can serve the client. If a service at either site goes down, the other server in that site has to handle twice as much traffic as a service at the other site. If dynamic weight is based on the number of active services, the site with both services active has twice the weight of the site with one service down and therefore receives twice as much traffic.

Alternatively, consider a configuration in which the services at the first site represent servers that are twice as powerful as servers at the second site. If dynamic weight is based on the weights assigned to the services, twice as much traffic can be sent to the first site as to the second.

Note: For details on assigning weights to load balancing services, see " Assigning Weights to Services".
As an illustration of how dynamic weight is calculated, consider a GSLB virtual server that has a GSLB service bound to it. The GSLB service represents a load balancing virtual server that in turn has two services bound to it. The weight assigned to the GSLB service is 3. The weights assigned to the two services are 1 and 2 respectively. In this example, when dynamic weight is set to:

- **Disabled**:The cumulative weight of the GSLB virtual server is the product of the dynamic weight (disabled = 1) and the weight of the GSLB service (3), so the cumulative weight is 3.
- **SERVICECOUNT**: The count is the sum of the number of services bound to the load balancing virtual servers corresponding to the GSLB service (2), and the cumulative weight is the product of the dynamic weight (2) and the weight of the GSLB service (3), which is 6.
- **SERVICEWEIGHT**: The dynamic weight is the sum of the number of services bound to the GSLB service (2), and the cumulative weight is the product of the dynamic weight (2) and the weight of the GSLB service (3), which is 6.

Note: Dynamic weights are not applicable when content switching virtual servers are configured.

To configure a GSLB virtual server to use dynamic weights by using the command line interface

At the command prompt, type:

set gslb vserver <name> -dynamicWeight SERVICECOUNT | SERVICEWEIGHT
Example

set gslb vserver vserver-GSLB-1 -dynamicWeight SERVICECOUNT

To set GSLB virtual server to use dynamic weights by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB virtual server for which you want to set dynamic weights (for example, vserver-GSLB-1).
3. Click Open.
4. On the Method and Persistence tab, under Method, select SERVICECOUNT or SERVICEWEIGHT from the Dynamic Weight list.
5. Click OK.

Changing the GSLB Method

Mar 22, 2012

Unlike traditional DNS servers that simply respond with the IP addresses of the configured servers, a NetScaler appliance configured for GSLB responds with the IP addresses of the services, as determined by the configured GSLB method. By default, the GSLB virtual server is set to the least connection method. If all GSLB services are down, the NetScaler responds with the IP addresses of all the configured GSLB services.

GSLB methods are algorithms that the GSLB virtual server uses to select the best-performing GSLB service. After the host name in the Web address is resolved, the client sends traffic directly to the resolved service IP address.

The NetScaler appliance provides the following GSLB methods:

- Round Robin
- Least Connections
- Least Response Time
- Least Bandwidth
- Least Packets
- Source IP Hash
- Custom Load
- Round Trip Time (RTT)
- Static Proximity

For GSLB methods to work with a remote site, either MEP must be enabled or explicit monitors must be bound to the remote services. If MEP is disabled, RTT, Least Connections, Least Bandwidth, Least Packets and Least Response Time methods default to Round Robin.

The Static Proximity and RTT load balancing methods are specific to GSLB.

Updated: 2013-11-11

For information about the Round Robin, Least Connections, Least Response Time, Least Bandwidth, Least Packets, Source IP Hash, or Custom Load method, see "Load Balancing."

To change the GSLB method by using the command line interface

At the command prompt, type:

set gslb vserver <name> -lbMethod GSLBMethod

Example

set gslb vserver Vserver-GSLB-1 -lbMethod ROUNDROBIN

To change the GSLB method by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the details pane, select a GSLB virtual server and click Open.
3. In the Configure GSLB Virtual Server dialog box, on the Method and Persistence tab, under Method, select a method from the Choose Method list.

4. Click OK, and verify that the method you selected appears under Details at the bottom of the screen.

Configuring Static Proximity

May 22, 2015

The static proximity method for GSLB uses an IP-address based static proximity database to determine the proximity between the client's local DNS server and the GSLB sites. The NetScaler appliance responds with the IP address of a site that best matches the proximity criteria.

If two or more GSLB sites at different geographic locations serve the same content, the NetScaler appliance maintains a database of IP address ranges and uses the database for decisions about the GSLB sites to which to direct incoming client requests.

For the static proximity method to work, you must either configure the NetScaler appliance to use an existing static proximity database populated through a location file or add custom entries to the static proximity database. After adding custom entries, you can set their location qualifiers. After configuring the database, you are ready to specify static proximity as the GSLB method.

This document includes the following information:

- Adding a Location File to Create a Static Proximity Database
- Adding Custom Entries to a Static Proximity Database
- Setting the Location Qualifiers
- Specifying the Proximity Method
- Synchronizing GSLB Static Proximity Database

Adding a Location File to Create a Static Proximity Database

Dec 22, 2014

A static proximity database is a UNIX-based ASCII file. Entries added to this database from a location file are called static entries. Only one location file can be loaded on a NetScaler appliance. Adding a new location file overrides the existing file. The number of entries in the static proximity database is limited by the configured memory in the NetScaler appliance.

The static proximity database can be created in the default format or in a format derived from commercially configured third party databases (such as www.maxmind.com and www.ip2location.com).

These databases vary in the details they provide. There is no strict enforcement of the database file format, except that the default file has format tags. The database files are ASCII files that use a comma as the field delimiter. There are differences in the structure of fields and the representation of IP addresses in the locations.

The format parameter describes the structure of the file to the NetScaler appliance. Specifying an incorrect value for the format option can corrupt the internal data.

Note: The default location of the database file is /var/netscaler/locdb, and on a high availability (HA) setup, an identical copy of the file must be present in the same location on both NetScaler appliances.
The following abbreviations are used in this section:

- **CSHN**. Short name of a country based on the country code standard of ISO-3166.
- **LCN**. Long name of the country.
- **RC**. Region code based on ISO-3166-2 (for US and Canada). The region code “FIPS-10-4” is used for the other regions.

Note: Some databases provide short country names according to ISO-3166 and long country names as well. The NetScaler uses short names when storing and matching qualifiers.
To create a static proximity database, log on to the UNIX shell of the NetScaler appliance and use an editor to create a file with the location details in one of the NetScaler-supported formats.

To add a static location file by using the command line interface

At the command prompt, type:

- add locationFile <locationFile> [-format <format>]
- show locationFile

Example

```
> add locationFile /var/nsmap/locdb/nsgeo1.0 -format netscaler
 Done
> show locationFile
Location File: /var/nsmap/locdb/nsgeo1.0
Format: netscaler
 Done
>
```

To add a static location file by using the configuration utility

1. Navigate to AppExpert > Location, click the **Static Database** tab.

2. Click **Add** to add a static location file.

You can view an imported location file database by using the View Database dialog box in the configuration utility. There is no NetScaler command line equivalent.

To view a static location file by using the configuration utility

1. Navigate to AppExpert > Location, click the **Static Database** tab.
2. Select a static location file, and from the **Action** list, click **View Database**.

To convert a location file into the netscaler format

By default, when you add a location file, it is saved in the netscaler format. You can convert a location file of other formats into the netscaler format.

Note: The nsmap option can be accessed only from the command line interface. The conversion is possible only into the netscaler format.
To convert the static database format, at the NetScaler command prompt, type the following command:

nsmap -f <inputFileFormat> -o <outputFileName> <inputFileName>

Example

```
nsmap -f ip-country-region-city -o nsfile.ns ip-country-region-city.cvs
```

Adding Custom Entries to a Static Proximity Database

Jan 22, 2015

Custom entries take precedence over static entries in the proximity database. You can add a maximum of 500 custom entries. For a custom entry, denote all omitted qualifiers with an asterisk (*) and, if qualifiers have a period or space in the name, enclose the parameter in double quotation marks. The first 31 characters are evaluated for each qualifier. You can also provide the longitude and latitude of the geographical location of the IP address-range for selecting a service with the static proximity GSLB method.

To add custom entries by using the command line interface

At the command prompt, type the following commands to add a custom entry to the static proximity database and verify the configuration:

- add location < IPfrom> < IPto> <preferredLocation> [-longitude <integer>[-latitude <integer>]]
- show location

Example

```
>add location 192.168.100.1 192.168.100.100 *.us.ca.mycity
>show location
```

Parameters for adding custom entries

IPfrom

First IP address in the range, in dotted decimal notation. This is a mandatory argument.

IPto

Last IP address in the range, in dotted decimal notation. This is a mandatory argument.

preferredLocation

String of qualifiers, in dotted notation, describing the geographical location of the IP address range. Each qualifier is more specific than the one that precedes it, as in continent.country.region.city.isp.organization. For example,"NA.US.CA.San Jose.ATT.citrix".

Note: A qualifier that includes a dot (.) or space ( ) must be enclosed in double quotation marks.

This is a mandatory argument. Maximum Length: 197

longitude

Numerical value, in degrees, specifying the longitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

Maximum value: 180

latitude

Numerical value, in degrees, specifying the latitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

Maximum value: 180

To add custom entries by using the configuration utility

Navigate to AppExpert > Location, click the **Custom Entries** tab, and add the custom entries.

Setting the Location Qualifiers

Nov 24, 2014

The database used to implement static proximity contains the location of the GSLB sites. Each location contains an IP address range and up to six qualifiers for that range. The qualifiers are literal strings and are compared in a prescribed order at run time. Every location must have at least one qualifier. The meaning of the qualifiers (context) is defined by the qualifier labels, which are user defined. The NetScaler has two built-in contexts:
Geographic context, which has the following qualifier labels:

- Qualifier 1 – “Continent”
- Qualifier 2 – “Country”
- Qualifier 3 – “State”
- Qualifier 4 – “City”
- Qualifier 5 – “ISP”
- Qualifier 6 – “Organization”

Custom entries, which have the following qualifier labels:

- Qualifier 1 – “Qualifier 1”
- Qualifier 2 – “Qualifier 2”
- Qualifier 3 – “Qualifier 3”
- Qualifier 4 – “Qualifier 4”
- Qualifier 5 – “Qualifier 5”
- Qualifier 6 – “Qualifier 6”

If the geographic context is set with no Continent qualifier, Continent is derived from Country. Even the built-in qualifier labels are based on the context, and the labels can be changed. These qualifier labels specify the locations mapped with the IP addresses used to make static proximity decisions.

To perform a static proximity-based decision, the NetScaler appliance compares the location attributes (qualifiers) derived from the IP address of the local DNS server resolver with the location attributes of the participating sites. If only one site matches, the appliance returns the IP address of that site. If there are multiple matches, the site selected is the result of a round robin on the matching GSLB sites. If there is no match, the site selected is a result of a round robin on all configured sites. A site that does not have any qualifiers is considered a match.

To set the location qualifiers by using the command line interface

At the command prompt, type:

set locationparameter -context <context> -q1label <string> [-q2label <string>] [-q3label <string>] [-q4label <string>] [-q5label <string>] [-q6label <string>]
Example

```
set locationparameter -context custom -q1label asia
```

To set the location qualifiers by using the configuration utility

1. Navigate to Traffic Management > GSLB > Location.
2. Click Location Parameters.
3. In the Context drop-down list, select the appropriate context (for example, Custom).
4. In the Qualifier Label -1 text box, type the qualifier (for example asia).

5. Click OK.

Specifying the Proximity Method

Nov 24, 2014

When you have configured the static proximity database, you are ready to specify static proximity as the GLSB method.

To specify static proximity by using the command line interface

At the command prompt, type the following commands to configure static proximity and verify the configuration:

- set gslb vserver <name> -lbMethod STATICPROXIMITY
- show gslb vserver <name>

Example

```
set gslb vserver Vserver-GSLB-1 -lbMethod STATICPROXIMITY
show gslb vserver
```

To specify static proximity by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB Virtual Server that you want to set to static proximity (for example, vserver-GSLB-1).
3. Click Open.
4. On the Method and Persistence tab, under Method, select Static Proximity from the Choose Method list.
5. Click OK.
6. Verify that the Details pane shows static proximity as the GSLB method.

Synchronizing GSLB Static Proximity Database

Apr 08, 2014

Synchronizing a global server load balancing (GSLB) static proximity database requires that one of the sites be identified as the master GSLB node. Any site in the topology can be designated as the master node. The rest of the GSLB nodes are automatically designated as slave nodes.

Synchronizing GSLB static proximity databases synchronizes the files in the /var/netscaler/locdb directory across the slave nodes. During the synchronization process, the master node fetches the running configuration from each of the slave nodes and compares it to the configuration on the master node. The master GSLB node uses the rsync program to synchronize the static proximity database across the slave nodes. To speed up the synchronization process, the rsync program makes only enough changes to eliminate the differences between the two files. The synchronization process cannot be rolled back.

The following example synchronizes Site2, which is a slave site, to master site Site1. The administrator enters the **sync gslb config** command on Site1:

```
sync gslb config -nowarn
Sync Time: Feb 24 2014 14:56:16
Retrieving local site info: ok
Retrieving all participating gslb sites info:
0 bytes in 0 blocks
ok
site1[Master]:
      Getting Config: ok
site2[Slave]:
      Syncing gslb static proximity database: ok
      Getting Config: ok
      Comparing config: ok
      Applying changes: ok
 Done
```

Configuring the Dynamic Method (RTT)

May 22, 2015

Dynamic round trip time (RTT) is a measure of time or delay in the network between the client's local DNS server and a data resource. To measure dynamic RTT, the NetScaler appliance probes the client's local DNS server and gathers RTT metric information. The appliance then uses this metric to make its load balancing decision. Global server load balancing monitors the real-time status of the network and dynamically directs the client request to the data center with the lowest RTT value.

When a client's DNS request for a domain comes to the NetScaler appliance configured as the authoritative DNS for that domain, the appliance uses the RTT value to select the IP address of the best performing site to send it as a response to the DNS request.

The NetScaler appliance uses different mechanisms, such as ICMP echo request / reply (PING), UDP, and TCP to gather the RTT metrics for connections between the local DNS server and participating sites. The appliance first sends a ping probe to determine the RTT. If the ping probe fails, a DNS UDP probe is used. If that probe also fails, the appliance uses a DNS TCP probe.

These mechanisms are represented on the Netscaler appliance as Load Balancing Monitors and are easily identified due to their use of the "ldns" prefix. The three monitors, in their default order, are:

- ldns-ping
- ldns-dns
- ldns-tcp

These monitors are built in to the appliance and are set to safe defaults, but may be customized just like any other monitor on the appliance.

The default order may also be changed by setting it explicitly as a GSLB parameter. For example, to set the order to be the DNS UDP query followed by the PING and then TCP, type the following command:

set gslb parameter -ldnsprobeOrder DNS PING TCP

Unless they have been customized, the NetScaler appliance performs UDP and TCP probing on port 53, however unlike regular load balancing monitors the probes need not be successful in order to provide valid RTT information. ICMP port unavailable messages, TCP Resets and DNS error responses, which would usually constitute a failure are all acceptable for calculating the RTT value.

Once the RTT data has been compiled, the Netscaler uses the proprietary metrics exchange protocol (MEP) to exchange RTT values between participating sites. After calculating RTT metrics, the appliance sorts the RTT values to identify the data center with the best (smallest) RTT metric."

If RTT information is not available (for example, when a client's local DNS server accesses the site for the first time), the NetScaler appliance selects a site by using the round robin method and directs the client to the site.

To configure the dynamic method, you configure the site's GSLB virtual server for dynamic RTT. You can also set the interval at which local DNS servers are probed to a value other than the default.

This document includes the following information:

- Configuring a GSLB Virtual Server for Dynamic RTT

Configuring a GSLB Virtual Server for Dynamic RTT

Updated: 2014-11-24

To configure a GSLB virtual server for dynamic RTT, you specify the RTT load balancing method.

The NetScaler appliance regularly validates the timing information for a given local server. If a change in latency exceeds the configured tolerance factor, the appliance updates its database with the new timing information and sends the new value to other GSLB sites by performing a MEP exchange. The default tolerance factor is 5 milliseconds (ms).

The RTT tolerance factor must be the same throughout the GSLB domain. If you change it for a site, you must configure identical RTT tolerance factors on all NetScaler appliances deployed in the GSLB domain.

To configure a GSLB virtual server for dynamic RTT by using the command line interface

At the command prompt, type:

set gslb vserver <name> -lbMethod RTT -tolerance <value>
Example

set gslb vserver Vserver-GSLB-1 -lbMethod RTT -tolerance 10

To configure a GSLB virtual server for dynamic RTT by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB Virtual server that you want to set to dynamic RTT (for example, vserver-GSLB-1).
3. Click Open.
4. On the Method and Persistence tab, under Method, select Dynamic Method (RTT) from the Choose Method list.
5. To change the tolerance factor, type the new value in the Tolerance (ms) text box.
6. Click OK.

Setting the Probing Interval of Local DNS Servers

Updated: 2014-11-24

The NetScaler appliance uses different mechanisms, such as ICMP echo request / reply (PING), TCP, and UDP to obtain RTT metrics for connections between the local DNS server and participating GSLB sites. By default, the appliance uses a ping monitor and probes the local DNS server every 5 seconds. The appliance then waits 2 seconds for the response and, if a response is not received in that time, it uses the TCP DNS monitor for probing.

However, you can modify the time interval for probing the local DNS server to accommodate your configuration.

To modify the probing interval by using the command line interface

At the command prompt, type:

set lb monitor <monitorName> <type> -interval <integer> <units> -resptimeout <integer> <units>

Example

```
set lb monitor monitor-HTTP-1 HTTP -interval 10 sec -resptimeout 5 sec
```

To modify the probing interval by using the configuration utility

- Navigate to Traffic Management > Load Balancing > Monitors.
- Select the monitor that you want to modify (for example, ping).
- Click Open.
- Click OK.

Configuring Persistent Connections

Jun 04, 2015

Persistence ensures that a series of client requests for a particular domain name is sent to the same data center instead of being load balanced. If persistence is configured for a particular domain, it takes precedence over the configured GSLB method. Persistence is useful for deployments that deal with e-commerce, such as shopping card usage, where the server needs to maintain the state of the connection to track the transaction. To maintain the state of connection, you must configure persistence on a virtual server. With persistence configured, NetScaler selects a data center to process a client request and forwards the IP address of the selected data center for all subsequent DNS requests. If the configured persistence applies to a site that is down, the NetScaler appliance uses a GSLB method to select a new site, and the new site becomes persistent for subsequent requests from the client.

The GSLB virtual server is responsible for DNS-based site persistence, and it controls the site persistence for a remote GSLB service. The NetScaler appliance supports persistence based on the source IP address or on HTTP cookies.

When you bring a physical service DOWN with a delay time, the physical service goes into the transition out of service (TROFS) state. Site persistence is supported as long as the service is in the TROFS state. That is, if the same client sends a request for the same service within the specified delay time after a service is marked TROFS, the same GSLB site (data center) services the request.

Note: If connection proxy is specified as the site persistence method and if you also want to configure persistence of the physical servers, do not configure SOURCEIP persistence. When the connection is proxied, an IP address owned by the NetScaler is used, and not the actual IP address of the client. Configure methods such as cookie persistence or rule-based persistence on the load balancing virtual server.
This document includes the following information:

- Configuring Persistence Based on Source IP Address
- Configuring Persistence Based on HTTP Cookies

Configuring Persistence Based on Source IP Address

Updated: 2014-11-24

With source-IP persistence, when a DNS request is received at a data center, the NetScaler appliance first looks for an entry in the persistence table and, if an entry for the local DNS server exists and the server mentioned in the entry is configured, the IP address of that server is sent as the DNS response.

For the first request from a particular client, the NetScaler appliance selects the best GSLB site for the request and sends its IP address to the client. Since persistence is configured for the source IP address of the client, all subsequent requests by that client or another local DNS server in the same IP subnet are sent the IP address of the GSLB site that was selected for the first request.

For source-IP address based persistence, the same set of persistence identifiers must be configured on the GSLB virtual servers in all data centers. A persistence identifier is a number used by the data centers to identify a particular GSLB virtual server. A cookie transmits the persistence identifier, enabling the NetScaler appliance to identify the domain so that it can forward all appropriate requests to the same domain. When persistence is enabled, the persistence information is also exchanged as part of metrics exchange.

For the NetScaler appliance to support persistence across sites, persistence must be enabled on the GSLB virtual servers of all participating sites. When you use source IP address persistence on the network identifier, you must configure a subnet mask. For any domain, persistence takes precedence over any other configured GSLB method.

To configure persistence based on source IP address by using the command line interface

At the command prompt, type:

set gslb vserver <name> -persistenceType (SOURCEIP | NONE) -persistenceId <positive_integer> [-persistMask <netmask>] – [timeout <mins>]

Example

```
set gslb vserver vserver-GSLB-1 -persistenceType SOURCEIP -persistenceId 23 -persistMask 255.255.255.255 –timeout 2
```

To configure persistence based on source IP address by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB virtual server whose method you want to change (for example, vserver-GSLB-1).
3. Click Open.
4. On the Method and Persistence tab, under Persistence, select SOURCEIP from the Persistence list and specify values for the following parameters as shown:
   - Time-out—timeout
   - Persistence Id—persistenceID
   - IPv4 Netmask or IPv6 Mask length—persistMask
5. Click OK.

Configuring Persistence Based on HTTP Cookies

Updated: 2014-11-26

The NetScaler appliance provides persistence at the HTTP-request level by using connection proxy and HTTP redirect. With these persistence methods, the appliance uses an HTTP cookie (known as a “site cookie”) to reconnect the client to the same server. The NetScaler inserts the site cookie in the first HTTP response.

The site cookie contains information about the selected GSLB service on which the client has a persistent connection. The cookie expiration is based on the cookie timeout configured on the NetScaler appliance. If the virtual server names are not identical on all the sites, you must use the persistence identifier. Cookies inserted are compliant with RFC 2109.

When the NetScaler appliance responds to a client DNS request by sending the IP address of the selected GSLB site, the client sends an HTTP request to that GSLB site. The physical server in that GSLB site adds a site cookie to the HTTP header, and connection persistence is in effect.

If the DNS entry in the client cache expires, and then the client sends another DNS query and is directed to a different GSLB site, the new GSLB site uses the site cookie present in the client request header to implement persistence. If the GSLB configuration at the new site uses connection-proxy persistence, the new site creates a connection to the GSLB site that inserted the site cookie, proxies the client request to the original site, receives a response from the original GSLB site, relays that response back to the client, and closes the connection. If the GSLB configuration uses HTTP redirect persistence, the new site redirects the request to the site that originally inserted the cookie.

Note: Connection proxy persistence can be configured only for local services. However, connection proxy persistence must be enabled on both local and remote GSLB services that are configured for the GSLB virtual server.
Connection proxy occurs when the following conditions are satisfied:

- Requests are sent from a domain participating in GSLB. The domain is obtained from the URL/Host header.
- Requests are sent from a local GSLB service whose public IP address matches the public IP address of an active service bound to the GSLB virtual server.
- The local GSLB service has connection proxy enabled.
- The request includes a valid cookie that contains the IP address of an active remote GSLB service.

If one of the conditions is not met, connection proxy does not occur, but a site cookie is added if the local GSLB service has connection proxy enabled AND:

- No site cookie is supplied; OR,
- The site cookie refers to an IP address that is not an active GSLB remote service; OR,
- The cookie refers to the IP address of the virtual server on which the request is received.

The following are the limitations of using connection proxy site cookies:

- Site cookies do not work for non-HTTP(S) protocols.
- If an HTTP request is sent to a back-up virtual server, the virtual server does not add a cookie.
- Site cookies do not work if SSL client authentication is required.
- At the local site, the statistics for a GSLB service on a remote site are not the same as the statistics recorded for that service at the remote site. At the local site, the statistics for a remote GSLB service are slightly higher than the statistics that the remote site records

for that same service.

Redirect persistence can be used only:

- For HTTP or HTTPS protocols.
- If the domain name is present in the request (either in the URL or in the HOST header), and the domain is a GSLB domain.
- When the request is received on a backup VIP or a GSLB local service that is in the down state.

To set persistence based on HTTP cookies by using the command line interface

At the command prompt, type:

set gslb service <serviceName> -sitePersistence (ConnectionProxy [-sitePrefix <prefix>] | HTTPredirect -sitePrefix <prefix>)

Example

```
set gslb service service-GSLB-1 -sitePersistence ConnectionProxy
set gslb service service-GSLB-1 -sitePersistence HTTPRedirect -sitePrefix vserver-GSLB-1
```

To set persistence based on cookies by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. In the GSLB Services pane, select the service that you want to configure for site persistence (for example, service-GSLB-1).
3. Click Open.
4. On the Advanced tab, under Site Persistence type, specify values for the following parameters as shown:
   - Site Persistence type—sitePersistence
   - Site Prefix—sitePrefix
5. Click OK.

Overriding Static Proximity Behavior by Configuring Preferred Locations

Mar 22, 2012

You might want to direct traffic from a local DNS (LDNS) server or network to a GSLB service other than the GSLB service that the static proximity method selects for that traffic. That is, you have a *preferred location* for that traffic. To override the static proximity method with preferred locations, you can do the following:

1. Configure a DNS action that consists of a list of preferred locations. For more information about configuring a DNS action, see Configuring a DNS Action.
2. Configure a DNS policy to identify the traffic arriving from the LDNS server or network for which you want to override static proximity, and apply the action in the policy.
3. Bind the policy to the global request bind point.

In the DNS action, you can configure a list of up to 8 preferred locations. The locations must be provided in the dotted qualifier notation, which is the notation in which you add custom locations to the static proximity database. The locations can include wildcards for qualifiers that you want to omit. For information about the dotted qualifier notation for locations, see Adding Custom Entries to a Static Proximity Database. When entering the preferred locations, you must enter them in the descending order of priority.

When a policy evaluates to TRUE, the NetScaler appliance matches the preferred locations, in priority order, with the locations of GSLB services. Matches are of the following two types:

- If all the non-wildcard qualifiers in a preferred location match the corresponding qualifiers in the location of a GSLB service, the match is considered a perfect match. For example, a GSLB service location of *.UK.*.* or Europe.UK.*.* is a perfect match for the preferred location *.UK.*.*.
- If only a subset of the non-wildcard qualifiers match, the match is considered a partial match. For example, a GSLB service location of Europe.EG is a partial match for the preferred location Europe.UK.

When a DNS policy evaluates to TRUE, the following algorithm is used to select a GSLB service:

1. The appliance evaluates the preferred location that has the highest priority and moves down the priority order until a perfect match is found between a preferred location and the location of a GSLB service.

   If a perfect match is found, the appliance checks whether the corresponding GSLB service is up. If it is up, it returns the IP address of the GSLB service in the DNS response. If multiple perfect matches are found (which can happen when one or more wildcards are used in a preferred location), the appliance checks the state of each of the corresponding GSLB services and load balances the GSLB services that are up.

2. If a perfect match is not found for any of the preferred locations, the appliance returns to the preferred location that has the highest priority and moves down the priority order until a partial match is found between a preferred location and the location of a GSLB service.

   If a partial match is found, the appliance checks whether the corresponding GSLB service is up. If it is up, it returns the IP address of the GSLB service in the DNS response. If multiple partial matches are found, the appliance checks the state of each of the corresponding GSLB services and load balances the GSLB services that are up.

3. If none of the perfect and partial matches are up, the appliance load balances all other available GSLB services.

In this way, the appliance implements a type of site affinity for traffic that matches the DNS policy.

Example

Consider a GSLB configuration that consists of the following eight GSLB services:

- Asia.IN
- Asia.JPN
- Asia.HK
- Europe.UK
- Europe.RU
- Europe.EG
- Africa.SD
- Africa.ZMB

Further consider the following DNS action and policy configuration:

```
> add dns action prefLoc11 GslbPrefLoc -preferredLocList "Asia.HK" "Europe.UK"
 Done
> add dns policy dnsPolPrefLoc "CLIENT.IP.SRC.MATCHES_LOCATION(\"*.ZMB.*.*\")" prefLoc11
 Done
```

When the appliance receives a request from the location *.ZMB.*.*, the preferred locations are evaluated as follows:

1. The appliance attempts to find a GSLB service whose location is a perfect match for Asia.HK, which is the preferred location that has the highest priority. It finds that the GSLB service at Asia.HK is a perfect match. If the GSLB service is up, it sends the client the IP address of the GSLB service.
2. If the GSLB service at Asia.HK is down, the appliance attempts to find a perfect match for the second preferred location, Europe.UK. It finds that the GSLB service at Europe.UK is a perfect match. If the GSLB service is up, it sends the client the IP address of the service.
3. If the GSLB service at Europe.UK is down, it returns to the preferred location that has the highest priority, Asia.HK, and looks for partial matches. For Asia.HK, it finds that Asia.IN and Asia.JPN are partial matches. If only one of the corresponding GSLB services is up, it sends the client the IP address of the service. If both locations are up, it load balances the two services.
4. If all partial matches for Asia.HK are down, the appliance looks for partial matches for Europe.UK. It finds that Europe.RU and Europe.EG are partial matches for the preferred location. If only one of the corresponding GSLB services is up, it sends the client the IP address of the service. If both locations are up, it load balances the two services.
5. If all partial matches for Europe.UK are down, the appliance load balances all other available GSLB services. In the current example, the appliance load balances Africa.SD and Africa.ZMB because the remaining six GSLB services have been found to be down.

Monitoring GSLB Services

May 22, 2015

When you bind a remote service to a GSLB virtual server, the GSLB sites exchange metric information, including network metric Information, which is the round-trip-time and persistence Information.

If a metric exchange connection is momentarily lost between any of the participating sites, the remote site is marked as DOWN and load balancing is performed on the remaining sites that are UP. When metric exchange for a site is DOWN, the remote services belonging to the site are marked DOWN as well.

The NetScaler appliance periodically evaluates the state of the remote GSLB services by using either MEP or monitors that are explicitly bound to the remote services. Binding explicit monitors to local services is not required, because the state of the local GSLB service is updated by default using the MEP. However, you can bind explicit monitors to a remote service. When monitors are explicitly bound, the state of the remote service is not controlled by the metric exchange.

By default, when you bind a monitor to a remote GSLB service, the NetScaler appliance uses the state of the service reported by the monitor. However, you can configure the NetScaler appliance to use monitors to evaluate services in the following situations:

- Always use monitors (default setting).
- Use monitors when MEP is DOWN.
- Use monitors when remote services and MEP are DOWN.

The second and third of the above settings enable the NetScaler to stop monitoring when MEP is UP. For example, in a hierarchical GSLB setup, a GSLB site provides the MEP information about its child sites to its parent site. Such an intermediate site may evaluate the state of the child site as DOWN because of network issues, though the actual state of the site is UP. In this case, you can bind monitors to the services of the parent site and disable MEP to determine the actual state of the remote service. This option enables you to control the manner in which the states of the remote services are determined.

To use monitors, first create them, and then bind them to GSLB services.

This document includes the following information:

- Adding or Removing Monitors
- Binding Monitors to a GSLB Service

Adding or Removing Monitors

Updated: 2014-11-24

To add a monitor, you specify the type and the port. You cannot remove a monitor that is bound to a service. You must first unbind the monitor from the service.

To add a monitor by using the command line interface

At the command prompt, type the following commands to create a monitor and verify the configuration:

- add lb monitor <monitorName> -type <monitorType> -destPort <portNumber>
- show lb monitor <monitorName>

Example

add lb monitor monitor-HTTP-1 -type HTTP -destPort 80
show lb monitor monitor-HTTP-1

To remove a monitor by using the command line interface

At the command prompt, type:

rm lb monitor <monitorName>

To add a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify values for the following parameters
   - Name*—name
   - Type*—type

   * A required parameter
4. On the Standard Parameters tab, in the Destination Port text box, type the destination port number (see “destPort” in the above parameter list).
5. Click Create, and then click Close.

Binding Monitors to a GSLB Service

Updated: 2014-11-24

Once you create monitors, you must bind them to GSLB services. When binding monitors to the services, you can specify a weight for the monitor. After binding one or more weighted monitors, you can configure a monitor threshold for the service. This threshold takes the service down if the sum of the bound monitor weights falls below the threshold value.

Note: In the configuration utility, you can set both the weight and the monitoring threshold at the same time that you bind the monitor. When using the command line, you must issue a separate command to set the service’s monitoring threshold.

To bind the monitor to the GSLB service by using the command line interface

At the command prompt, type:

bind monitor <name> <serviceName> [-state (**Enabled** | **Disabled**)] -weight <positiveInteger>
Example

bind monitor monitor-HTTP-1 service-GSLB-1 -state enabled -weight 2

To set the monitoring threshold for a GSLB service by using the command line interface

At the command prompt, type:

set gslb service <ServiceName> -monThreshold <PositiveInteger>
Example

set gslb service service-GSLB-1 -monThreshold 9

To bind the monitor to the GSLB service by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. The details pane, select the service to which you want to bind the monitor (for example, select service-GSLB-1).
3. Click Open.
4. In the Configure GSLB Service dialog box, on the Monitors tab, select the monitor that you want to bind to the service (for example, monitor-HTTP-1).
5. Click Add.
6. In the Configured table, you can select the newly assigned monitor and enter a new weight value.
7. To enable the monitor, make sure the State check box is selected.
8. Repeat the preceding steps to add additional monitors.
9. In the Monitor Threshold text box, you can enter a threshold value.
10. Click OK.

Monitoring GSLB Sites

Nov 24, 2014

The NetScaler appliance uses MEP or monitors to determine the state of the GSLB sites. You can configure a GSLB site to always use monitors (the default), use monitors when MEP is down, or use monitors when both the remote service and MEP are down. In the latter two cases, the NetScaler appliance stops monitoring when MEP returns to the UP state.

To configure monitor triggering by using the command line interface

At the command prompt, type:

set gslb site <siteName> –triggerMonitor (ALWAYS | MEPDOWN | MEPDOWN_SVCDOWN)

Example

```
> set gslb site Site-GSLB-North-America –triggerMonitor Always
 Done
```

To configure monitor triggering by using the configuration utility

1. Navigate to Traffic Management > GSLB > Sites.
2. In the details pane, select the site, and then click Open.
3. In the Configure GSLB Site dialog box, in the Trigger Monitors drop-down list, select an option for when to trigger monitoring.
4. Click OK.

Protecting the GSLB Setup Against Failure

May 22, 2015

You can protect your GSLB setup against failure of a GSLB site or a GSLB virtual server by configuring a backup GSLB virtual server, configuring the NetScaler appliance to respond with multiple IP addresses, or configuring a Backup IP address for a GSLB domain. You can also divert excess traffic to a backup virtual server by using spillover.

This document includes the following information:

- Configuring a Backup GSLB Virtual Server
- Configuring a GSLB Setup to Respond with Multiple IP Addresses
- Configuring a GSLB Virtual Server to Respond with an Empty Address Record When DOWN
- Configuring a Backup IP Address for a GSLB Domain
- Diverting Excess Traffic to a Backup Virtual Server

Configuring a Backup GSLB Virtual Server

Updated: 2015-05-04

Configuring a backup entity for a GSLB virtual server ensures that DNS traffic to a site is not interrupted if the GSLB virtual server goes down. The backup entity can be another GSLB virtual server, or it can be a backup IP address. With a backup entity configured, if the primary GSLB virtual server goes down, the backup entity handles DNS requests. To specify what should happen when the primary GSLB virtual server comes back up again, you can configure the backup entity to continue handling traffic until you manually enable the primary virtual server to take over (using the disablePrimaryOnDown option), or you can configure a timeout period after which the primary takes over.

If you configure both the timeout and the disablePrimaryOnDown option for the backup entity, the backup session time-out takes precedence over the disablePrimaryOnDown setting.

To configure a backup GSLB virtual server by using the command line interface

At the command prompt, type the following commands to configure a GSLB virtual server as a backup virtual server and verify the configuration:

- set gslb vserver <name> -backupVServer <name> [-backupSessionTimeout <timeoutValue>] [-disablePrimaryOnDown (ENABLED | DISABLED)]
- show gslb vserver <name>

Example

```
set gslb vserver vserver-GSLB-1 -backupVServer vserver-GSLB-2 -backupSessionTimeout 3 -disablePrimaryOnDown ENABLED
show gslb vserver vserver-GSLB-1
```

To set GSLB virtual server as a backup virtual server by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB virtual server for which you want to configure a backup virtual server (for example, vserver-GSLB-1).
3. Click Open.
4. On the Advanced tab, specify values for the following parameters as shown:
   - Backup VServer—backupVServer
   - Backup Session Time-out (mins)—backupSessionTimeout
   - Disable Primary When Down—disablePrimaryOnDown
5. Click OK.

Configuring a GSLB Setup to Respond with Multiple IP Addresses

Updated: 2014-11-24

A typical DNS response contains the IP address of the best performing GSLB service. However, if you enable multiple IP response (MIR), the

NetScaler appliance sends the best GSLB service as the first record in the response and adds the remaining active services as additional records. If MIR is disabled (the default), the NetScaler appliance sends the best service as the only record in the response.

To configure a GSLB virtual server for multiple IP responses by using the command line interface

At the command prompt, type the following commands to configure a GSLB virtual server for multiple IP responses and verify the configuration:

- set gslb vserver<name> -MIR (ENABLED | DISABLED)
- show gslb vserver <name>

Example

```
set gslb vserver vserver-GSLB-1 -MIR ENABLED
 show gslb vserver <vserverName>
```

To set a GSLB virtual server for multiple IP responses by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB virtual server for which you want to configure a backup virtual server (for example, vserver-GSLB-1).
3. Click Open.
4. On the Advanced tab, under When this VServer is “UP,” select the Send all “active” service IP in response (MIR) check box.
5. Click OK.

Configuring a GSLB Virtual Server to Respond with an Empty Address Record When DOWN

Updated: 2014-11-24

A DNS response can contain either the IP address of the requested domain or an answer stating that the IP address for the domain is not known by the DNS server, in which case the query is forwarded to another name server. These are the only possible responses to a DNS query.

When a GSLB virtual server is disabled or in a DOWN state, the response to a DNS query for the GSLB domain bound to that virtual server contains the IP addresses of all the services bound to the virtual server. However, you can configure the GSLB virtual server to in this case send an empty down response (EDR). When this option is set, a DNS response from a GSLB virtual server that is in a DOWN state does not contain IP address records, but the response code is successful. This prevents clients from attempting to connect to GSLB sites that are down.

Note: You must configure this setting for each virtual server to which you want it to apply.

To configure a GSLB virtual server for empty down responses by using the command line interface

At the command prompt, type:

set gslb vserver<name> -EDR (ENABLED | DISABLED)
Example

```
> set gslb vserver vserver-GSLB-1 -EDR ENABLED
 Done
```

To set a GSLB virtual server for empty down responses by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB virtual server for which you want to configure a backup virtual server (for example, vserver-GSLB-1).
3. Click Open.
4. On the Advanced tab, under When this VServer is “Down,” select the Do not send any service’s IP address in response (EDR) check box.
5. Click OK.

Configuring a Backup IP Address for a GSLB Domain

Updated: 2014-11-24

You can configure a backup site for your GSLB configuration. With this configuration in place, if all of the primary sites go DOWN, the IP address of the backup site is provided in the DNS response.

Typically, if a GSLB virtual server is active, that virtual server sends a DNS response with one of the active site IP addresses as selected by the configured GSLB method. If all the configured primary sites in the GSLB virtual server are inactive (in the DOWN state), the authoritative domain name system (ADNS) server or DNS server sends a DNS response with the backup site's IP address.

Note: When a backup IP address is sent, persistence is not honored.

To set a backup IP address for a domain by using the command line interface

At the command prompt, type the following commands to set a backup IP address and verify the configuration:

- set gslb vserver <name> -domainName <string> -backupIP <IPAddress>
- show gslb vserver <name>

Example

```
set gslb vserver vserver-GSLB-1 -domainName www.abc.com -backupIP 10.102.29.66
show gslb vserver vserver-GSLB-1
```

To set a backup IP address for a domain by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the GSLB Virtual Servers pane, select the GSLB virtual server to which you want to bind the backup domain (for example, vserver-GSLB-1).
3. Click Open.
4. On the Domains tab, select a domain and click Open.
5. In the Configure GSLB Domain dialog box, in the Backup IP text box, type the IP address of the backup domain.
6. Click OK.

Diverting Excess Traffic to a Backup Virtual Server

Updated: 2014-11-24

Once the number of connections to a primary GSLB virtual server exceeds the configured threshold value, you can use the spillover option to divert new connections to a backup GSLB virtual server. This threshold value can be calculated dynamically or set manually. Once the number of connections to the primary virtual server drops below the threshold, the primary GSLB virtual server resumes serving client requests.

You can configure persistence with spillover. When persistence is configured, new clients are diverted to the backup virtual server if that client is not already connected to a primary virtual server. When persistence is configured, connections that were diverted to the backup virtual server are not moved back to the primary virtual server after the number of connections to the primary virtual server drops below the threshold. Instead, the backup virtual server continues to process those connections until they are terminated by the user. Meanwhile, the primary virtual server accepts new clients.

The threshold can be measured either by the number of connections or by the bandwidth.

If the backup virtual server reaches the configured threshold and is unable to take any additional load, the primary virtual server diverts all requests to the designated redirect URL. If a redirect URL is not configured on the primary virtual server, subsequent requests are dropped.

The spillover feature prevents the remote backup GSLB service (backup GSLB site) from getting flooded with client requests when the primary GSLB virtual server fails. This occurs when a monitor is bound to a remote GSLB service, and the service experiences a failure that causes its state to go DOWN. The monitor continues to keep the state of the remote GSLB service UP, however, because of the spillover feature.

As part of the resolution to this problem, two states are maintained for a GSLB service, the primary state and effective state. The primary state is the state of the primary virtual server and the effective state is the cumulative state of the virtual servers (primary and backup chain). The effective state is set to UP if any of the virtual servers in the chain of virtual servers is UP. A flag that indicates that the primary VIP has reached the threshold is also provided. The threshold can be measured by either the number of connections or the bandwidth.

A service is considered for GSLB only if its primary state is UP. Traffic is directed to the backup GSLB service only when all the primary virtual servers are DOWN. Typically, such deployments will have only one backup GSLB service.

Adding primary and effective states to a GSLB service has the following effects:

- When source IP persistence is configured, the local DNS is directed to the previously selected site only if the primary virtual server on the selected site is UP and below threshold. Persistence can be ignored in the round robin mode.
- If cookie-based persistence is configured, client requests are redirected only when the primary virtual server on the selected site is UP.
- If the primary virtual server has reached its saturation and the backup VIP(s) is absent or down, the effective state is set to DOWN.

- If external monitors are bound to an HTTP-HTTPS virtual server, the monitor decides the primary state.
- If there is no backup virtual server to the primary virtual server and the primary virtual server has reached its threshold, the effective state is set to DOWN.

To configure a backup GSLB virtual server by using the command line interface

At the command prompt, type the following commands to configure a backup GSLB virtual server and verify the configuration:

- set gslb vserver <name> -soMethod <method> -soThreshold <threshold> -soPersistence (**ENABLED** | **DISABLED**) -soPersistenceTimeout <timeout>
- show gslb vserver <name>

Example

```
set gslb vserver Vserver-GSLB-1 -soMethod CONNECTION -soThreshold 1000 -soPersistence ENABLED -soPersistenceTimeout 2
show gslb vserver Vserver-GSLB-1
```

To configure a backup GSLB virtual server by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. In the details pane, select the virtual server that you want to configure as a backup (for example, Vserver-LB-1), and then click Open.
3. On the Advanced tab, under Spillover, specify values for the following parameters:
   - Method— soMethod
   - Threshold— soThreshold
   - Persistence Time-out (min) — soPersistenceTimeout
4. Select the Persistence check box.
5. Click OK.

Managing Client Connections

May 22, 2015

To facilitate management of client connections, you can enable delayed cleanup of connections to the virtual server. You can then manage local DNS traffic by configuring DNS policies.

This document includes the following information:

- Enabling Delayed Cleanup of Virtual Server Connections
- Managing Local DNS Traffic by Using DNS Policies
- Adding DNS Views

Enabling Delayed Cleanup of Virtual Server Connections

Updated: 2014-11-24

The state of a virtual server depends on the states of the services bound to it, and the state of each service depends on the monitors bound to it. If a server is slow or down, the monitoring probes time out and the service that represents the server is marked as DOWN. A virtual server is marked as DOWN only when all services bound to it are marked as DOWN. You can configure services and virtual servers to either terminate all connections when they go down, or allow the connections to go through. The latter setting is for situations in which a service is marked as DOWN because of a slow server.

When you configure the down state flush option, the NetScaler appliance performs a delayed cleanup of connections to a GSLB service that is down.

To enable delayed cleanup of virtual server connections by using the command line interface

At the command prompt, type the following commands to configure delayed connection cleanup and verify the configuration:

- set gslb service <name> -downStateFlush (ENABLED | DISABLED)
- show gslb service <name>

Example

```
> set gslb service Service-GSLB-1 -downStateFlush ENABLED
 Done
> show gslb service Service-GSLB-1
 Done
```

To enable delayed cleanup of virtual server connections by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. In the GSLB Services pane, select the service (for example, service-GSLB-1), and then click Open.
3. On the Advanced tab, select the Down state flush check box.
4. Click OK.

Managing Local DNS Traffic by Using DNS Policies

Updated: 2015-05-22

You can use DNS policies to implement site affinity by directing traffic from the IP address of a local DNS resolver or network to a predefined target GSLB site. This is configured by creating DNS policies with DNS expressions and binding the policies globally on the NetScaler appliance.

This document includes the following information:

- DNS Expressions
- Configuring DNS Actions
- Configuring DNS Policies
- Binding DNS Policies

DNS Expressions

Updated: 2013-07-18

The NetScaler appliance provides certain predefined DNS expressions that can be used for configuring actions specific to a domain. Such actions can, for example, drop certain requests, select a specific view for a specific domain, or redirect certain requests to a specific location.

These DNS expressions (also called *rules*) are combined to create DNS policies that are then bound globally on the NetScaler appliance.

Following is the list of predefined DNS qualifiers available on the NetScaler appliance:

- CLIENT.UDP.DNS.DOMAIN.EQ("domainname")
- CLIENT.UDP.DNS.IS_AREC
- CLIENT.UDP.DNS.IS_AAAAREC
- CLIENT.UDP.DNS.IS_SRVREC
- CLIENT.UDP.DNS.IS_MXREC
- CLIENT.UDP.DNS.IS_SOAREC
- CLIENT.UDP.DNS.IS_PTRREC
- CLIENT.UDP.DNS.IS_CNAME
- CLIENT.UDP.DNS.IS_NSREC
- CLIENT.UDP.DNS.IS_ANYREC

The CLIENT.UDP.DNS.DOMAIN DNS expression can be used with string expressions. If you are using domain names as part of the expression, they must end with a period (.). For example, CLIENT.UDP.DNS.DOMAIN.ENDSWITH("abc.com.")

To create an expression by using the configuration utility

1. Click the icon next to the Expression text box. Click Add. (Leave the Flow Type and Protocol drop-down list boxes empty.) Follow these steps to create a rule.
2. In the Qualifier box, select a qualifier (for example, LOCATION).
3. In the Operator box, select an operator (for example, ==).
4. In the Value box, type a value (for example, Asia, Japan....).
5. Click OK. Click Create and click Close. The rule is created.
6. Click OK.

Configuring DNS Actions

Updated: 2014-11-24

A DNS policy includes the name of a DNS action to be performed when the policy rule evaluates to TRUE. A DNS action can do one of the following:

- Send the client an IP address for which you have configured a DNS view. For more information about DNS views, see Adding DNS Views.
- Send the client the IP address of a GSLB service after referring to a list of preferred locations that overrides static proximity behavior. For more information about preferred locations, see Overriding Static Proximity Behavior by Configuring Preferred Locations.
- Send the client a specific IP address as determined by the evaluation of the DNS query or response (DNS response rewrite).
- Forward a request to the name server without performing a lookup in the appliance's DNS cache.
- Drop a request.

You cannot create a DNS action for dropping a DNS request or for bypassing the DNS cache on the appliance. If you want to drop a DNS request, use the built-in action, dns_default_act_Drop. If you want to bypass the DNS cache, use the built-in action, dns_default_act_Cachebypass. Both actions are available along with custom actions in the Create DNS Policy and the Configure DNS Policy dialog boxes. These built-in actions cannot be modified or removed.

To configure a DNS action by using the command line interface

At the command prompt, type the following commands to configure a DNS action and verify the configuration:

- add dns action <actionName> <actionType> (-IPAddress <ip_addr | ipv6_addr> ... | -viewName <string> | -preferredLocList <string> ...) [-TTL <secs>]
- show dns action [<actionName>]

Examples

Example 1: Configuring DNS Response Rewrite. The following DNS action sends the client a preconfigured IP address when the policy to which the action is bound evaluates to true:

```
> add dns action dns_act_response_rewrite Rewrite_Response -IPAddress 192.0.2.20 192.0.2.56 198.51.100.10
 Done
> show dns action dns_act_response_rewrite
1) ActionName:  dns_act_response_rewrite ActionType:  Rewrite_Response TTL:  3600          IPAddress:  192.0.2.20 192.0.2.56 198.51.100.10
 Done
```

Example 2: Configuring a DNS-View Based Response. The following DNS action sends the client an IP address for which you have configured a DNS view:

```
> add dns action send_ip_from_view_internal_ip ViewName -viewName view_internal_ip
 Done
> show dns action send_ip_from_view_internal_ip
1) ActionName:  send_ip_from_view_internal_ip ActionType:  ViewName   ViewName:  view_internal_ip
 Done
```

Example 3: Configuring a Response Based on a Preferred Location List. The following DNS action sends the client the IP address that corresponds to the preferred location that it selects from the specified list of locations:

```
> add dns action send_preferred_location GslbPrefLoc -preferredLocList NA.tx.ns1.*.*.* NA.tx.ns2.*.*.* NA.tx.ns3.*.*.*
 Done
```

```
> show dns action  send_preferred_location
1) ActionName:  send_preferred_location ActionType:  GslbPrefLoc PreferredLocList:  "NA.tx.ns1.*.*.*" "NA.tx.ns2.*.*.*" "NA.tx.ns3.*.*.*"
 Done
```

To configure a DNS action by using the NetScaler configuration utility

1. Navigate to Traffic Management > DNS > Actions.
2. In the details pane, do one of the following:
   - To create a DNS action, click Add.
   - To modify a DNS action, select the DNS action that you want to modify, and then click Open.
3. In the Create DNS Action or Configure DNS Action dialog box, set the following parameters:
   - Action Name (cannot be changed for an existing DNS action)
   - Type (cannot be changed for an existing DNS action)
     To set the Type parameter, do one of the following:
     - To create a DNS action that is associated with a DNS view, select View Name. Then, from the View Name list, select the DNS view that you want to use in the action.
     - To create a DNS action with a preferred location list, select Preferred Location List. In Preferred Location, enter a location, and then click Add. Add as many DNS locations as you want.
     - To configure a DNS action for rewriting a DNS response on the basis of policy evaluation, select Rewrite Response. In IP Address, enter an IP address, and then click Add. Add as many IP addresses as you want.
   - TTL (applicable only to the Rewrite Response action type)

Configuring DNS Policies

Updated: 2014-11-24

DNS policies operate on a location database that uses static and custom IP addresses. The attributes of the incoming local DNS request are defined as part of an expression, and the target site is defined as part of a DNS policy. While defining actions and expressions, you can use a pair of single quotation marks (‘’) as a wildcard qualifier to specify more than one location. When a DNS policy is configured and a GSLB request is received, the custom IP address database is first queried for an entry that defines the location attributes for the source:

- When a DNS query comes from an LDNS, the characteristics of the LDNS are evaluated against the configured policies. If they match, an appropriate action (site affinity) is executed. If the LDNS characteristics match more than one site, the request is load balanced between the sites that match the LDNS characteristics.
- If the entry is not found in the custom database, the static IP address database is queried for an entry, and if there is a match, the above policy evaluation is repeated.
- If the entry is not found in either the custom or static databases, the best site is selected and sent in the DNS response on the basis of the configured load balancing method.

The following restrictions apply to DNS policies created on the NetScaler appliance.

- A maximum of 64 policies are supported.
- DNS policies are global to the NetScaler and cannot be applied to a specific virtual server or domain.
- Domain or virtual server specific binding of policy is not supported.

You can use DNS policies to direct clients that match a certain IP address range to a specific site. For example, if you have a GSLB setup with multiple GSLB sites that are separated geographically, you can direct all clients whose IP address is within a specific range to a particular data center.

Both TCP-based and UDP-based DNS traffic can be evaluated. Policy expressions are available for UDP-based DNS traffic on the server and for both UDP-based DNS traffic and TCP-based DNS traffic on the client side. Additionally, you can configure expressions to evaluate queries and responses that involve only the following DNS question types (or QTYPE values):

- A
- AAAA
- NS
- SRV
- PTR
- CNAME
- SOA
- MX
- ANY

The following response codes (RCODE values) are also supported:

- NOERROR - No error
- FORMERR - Format error
- SERVFAIL - Server failure
- NXDOMAIN - Non-existent domain
- NOTIMP - Query type not implemented
- REFUSED - Query refused

You can configure expressions to evaluate DNS traffic. A DNS expression begins with the DNS.REQ or DNS.RES prefixes. Functions are available for evaluating the

queried domain, the query type, and the carrier protocol. For more information about DNS expressions, see "Expressions for Evaluating a DNS Message and Identifying Its Carrier Protocol" in "Policy Configuration and Reference".

To add a DNS policy by using the command line interface

At the command prompt, type the following commands to create a DNS policy and verify the configuration:

- add dns policy <name> <rule> <actionName>
- show dns policy <name>

Example

```
> add dns policy policy-GSLB-1 'CLIENT.UDP.DNS.DOMAIN.EQ(\"domainname\")' my_dns_action
 Done
> show dns policy policy-GSLB-1
 Name: policy-GSLB-1
 Rule: CLIENT.UDP.DNS.DOMAIN.EQ("domainname")
 Action Name: my_dns_action
 Hits: 0
 Undef Hits: 0

 Done
```

To remove a configured DNS policy by using the command line interface

At the command prompt, type:

rm dns policy <name>

To configure a DNS policy by using the NetScaler configuration utility

1. Navigate to Traffic Management > DNS > Policies.
2. In the details pane, do one of the following:
   - To create a DNS policy, click Add.
   - To modify a DNS policy, select the DNS policy, and then click OK.
3. In the Create DNS Policy or Configure DNS Policy dialog box, set the following parameters:
   - Policy Name (cannot be changed for an existing policy)
   - Action
   - Expression

     To specify an expression, do the following:
     1. Click Add, and then, in the drop-down box that appears, select the expression element with which you want to begin the expression. A second list appears. The list contains a set of expression elements that you can use immediately after the firs expression element.
     2. In the second list, select the expression element that you want, and then enter a period.
     3. After each selection, if you enter a period, the next set of valid expression elements appear in a list. Select expression elements and fill in arguments to functions until you have the expression you want.
4. Click Create or OK, and then click Close.

Binding DNS Policies

Updated: 2013-08-29

DNS policies are bound globally on the NetScaler appliance and are available for all configured GSLB virtual servers. Even though DNS policies are globally bound, policy execution can be limited to a specific GSLB virtual server by specifying the domain in the expression.

Note: Even though the bind dns global command accepts REQ_OVERRIDE and RES_OVERRIDE as valid bind points, those bind points are redundant, because DNS policies can be bound only globally. Bind your DNS policies only to the REQ_DEFAULT and RES_DEFAULT bind points.

To bind a DNS policy globally by using the command line interface

At the command prompt, type the following commands to bind a DNS policy globally and verify the configuration:

- bind dns global <policyName> <priority> [-gotoPriorityExpression <string>] [-type <type>]
- show dns global -type <type>

Example

```
> bind dns global policy-GSLB-1 10 -gotoPriorityExpression END
 Done
> show dns global -type REQ_DEFAULT
1) Policy Name: policy-GSLB-1
 Priority: 10
```

```
GotoPriorityExpression: END

Done
```

To bind a DNS policy globally by using the configuration utility

1. Navigate to Traffic Management > DNS > Policies.
2. In the details pane, click Global Bindings.
3. In the Bind/Unbind DNS Policy(s) to Global dialog box, click Insert Policy.
4. In the Policy Name column, select, from the list, the policy that you want to bind. Alternatively, in the list, click New Policy, and then create a DNS policy by setting parameters in the Create DNS Policy dialog box.
5. To modify a policy that is already bound globally, click the name of the policy, and then click Modify Policy. Then, in the Configure DNS Policy dialog box, modify the policy, and then click OK.
6. To unbind a policy, click the name of the policy, and then click Unbind Policy.
7. To modify the priority assigned to a policy, double-click the priority value, and then enter a new value.
8. To regenerate assigned priorities, click Regenerate Priorities. The priority values are modified to begin at 100, with increments of 10, without affecting the order of evaluation.
9. Click OK.

To view the global bindings of a DNS policy by using the command line interface

At the command prompt, type:

show dns global

To view the global bindings of a DNS policy by using the configuration utility

1. Navigate to Traffic Management > DNS > Policies.
2. In the details pane, click Global Bindings. The global bindings of all DNS policies appear in this dialog box.

Adding DNS Views

Updated: 2014-11-24

You can configure DNS views to identify various types of clients and provide an appropriate IP address to a group of clients who query for the same GSLB domain. DNS views are configured by using DNS policies that select the IP addresses sent back to the client.

For example, if you have configured GSLB for your company's domain and have the server hosted in your company's network, clients querying for the domain from within your company's internal network can be provided with the server's internal IP address instead of the public IP address. Clients that query DNS for the domain from the Internet, on the other hand, can be provided the domain's public IP address.

To add a DNS view, you assign it a name of up to 31 characters. The leading character must be a number or letter. The following characters are also allowed: @ _ - . (period) : (colon) # and space ( ). After adding the view, you configure a policy to associate it with clients and a part of the network, and you bind the policy globally. To configure and bind a DNS policy, see Configuring DNS Policies and Binding DNS Policies.

To add a DNS view by using the command line interface

At the command prompt, type the following commands to create a DNS view and verify the configuration:

- add dns view <viewName>
- show dns view <viewName>

Example

```
add dns view PrivateSubnet
show dns view PrivateSubnet
```

To remove a DNS view by using the command line interface

At the command prompt, type:

rm dns view <viewName>

To add a DNS view by using the configuration utility

1. Navigate to Traffic Management > DNS > Views.
2. In the details pane, click Add.
3. In the Create DNS view dialog box, in the Name text box, enter the name of the DNS view.
4. Click Create, and then click Close. The DNS view that you created appears in the Views pane.

For details on how to create a DNS policy, see Configuring DNS Policies and for details on how to bind DNS policies globally, see Binding DNS Policies.

Configuring GSLB for Disaster Recovery

May 22, 2015

Disaster recovery capability is critical, because downtime is costly. A NetScaler appliance configured for GSLB forwards traffic to the least-loaded or the best-performing data center. This configuration, referred to as an active-active setup, not only improves performance, but also provides immediate disaster recovery by routing traffic to other data centers if a data center that is part of the setup goes down. Alternatively, you can configure an active-standby GSLB setup for disaster recovery only.

This document includes the following information:

- Configuring GSLB for Disaster Recovery in an Active-Standby Data Center Setup
- Configuring for Disaster Recovery in an Active-Active Data Center Setup
- Configuring for Disaster Recovery with Weighted Round Robin
- Configuring for Disaster Recovery with Data Center Persistence

Configuring GSLB for Disaster Recovery in an Active-Standby Data Center Setup

Updated: 2014-11-24

A conventional disaster recovery setup includes an active data center and a standby data center. The standby data center is a remote site. When a failover occurs as a result of a disaster event that causes the primary active data center to be inactive, the standby data center becomes operational.
Configuring disaster recovery in an active-standby data-center setup consists of the following tasks.

- Create the active data center.
  - Add a local GSLB site.
  - Add a GSLB vserver, which represents the active data center.
  - Bind the domain to the GSLB virtual server.
  - Add gslb services and bind the services to active GSLB virtual server.
- Create the standby data center.
  - Add a remote gslb site.
  - Add a gslb vserver, which represents standby data center.
  - Add gslb services which represents standby data center and bind the services to the standby gslb vserver.
  - Designate the standby data center by configuring the standby GSLB virtual server as the backup virtual server for the active GSLB virtual server.

Once you have configured the primary data center, replicate the configuration for the backup data center and designate it as the standby GSLB site by designating a GSLB virtual server at that site as the backup virtual server.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

To designate the standby GSLB site by using the command line interface

At both the active site and the remote site, at the command prompt, type:

set gslb vserver <name> -backupVserver <string>
Example

```
set gslb vserver vserver-GSLB-1 -backupVServer vserver-GSLB-2
```

To configure the standby site by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. Select the GSLB virtual server for the primary site and click Open.
3. In the Configure GSLB Virtual Server dialog box, on the Advanced tab, in the Backup VServer drop-down list box, select a backup virtual server.
4. Click OK.

By default, once the primary virtual server becomes active, it starts receiving traffic. However, if you want the traffic to be directed to the backup virtual server even after the primary virtual server becomes active, use the 'disable primary on down' option.

Configuring for Disaster Recovery in an Active-Active Data Center Setup

An active-active GSLB deployment, in which both GSLB sites are active, removes any risk that may arise in having a standby data center. With such a setup, web or application content can be mirrored in geographically separate locations. This ensures that data is consistently available at each distributed data center.

To configure GSLB for disaster recovery in an active-active data center set up, you must first configure the basic GSLB setup on the first data center and then configure all other data centers.

First create at least two GSLB sites. Then, for the local site, create GSLB a virtual server and GSLB services and bind the services to the virtual servers. Then create ADNS services and bind the domain for which you are configuring GSLB to the GSLB virtual server in the local site. Finally, at the local site, create a load balancing virtual server with the same virtual server IP address as the GSLB service.

Once you have configured the first data center, replicate the configuration for other data centers part of the setup.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

Configuring for Disaster Recovery with Weighted Round Robin

Updated: 2014-11-24

When you configure GSLB to use the weighted round robin method, weights are added to the GSLB services and the configured percentage of incoming traffic is sent to each GSLB site. For example, you can configure your GSLB setup to forward 80 percent of the traffic to one site and 20 percent of the traffic to another. After you do this, the NetScaler appliance will send four requests to the first site for each request that it sends to the second.

To set up the weighted round robin method, first create two GSLB sites, local and remote. Next, for the local site create a GSLB virtual server and GSLB services, and bind the services to the virtual servers. Configure the GSLB method as round robin. Next, create ADNS services and bind the domain for which you are configuring GSLB to the GSLB virtual server. Finally, create a load balancing virtual server with the same virtual server IP address as the GSLB service.

Each service that represents a physical server in the network has weights associated with it. Therefore the GSLB service is assigned a dynamic weight that is the sum of weights of all services bound to it. Traffic is then split between the GSLB services based on the ratio of the dynamic weight of the particular service to the total weight. You can also configure individual weights for each GSLB service instead of the dynamic weight.

If the services do not have weights associated with them, you can configure the GSLB virtual server to use the number of

services bound to it to calculate the weight dynamically.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

Once you configure a basic GSLB setup, you must configure the weighted round robin method such that the traffic is split between the configured GSLB sites according to the weights configured for the individual services.

To configure a virtual server to assign weights to services by using the command line interface

At the command prompt, type one of the following commands, depending upon whether you want to create a new load balancing virtual server or configure an existing one:

- add lb vserver <name>@ -weight <WeightValue> <ServiceName>
- set lb vserver <name>@ -weight <WeightValue> <ServiceName>

Example

```
add lb vserver Vserver-LB-1 -weight 4 Service-HTTP-1
set lb vserver Vserver-LB-1 -weight 4 Service-HTTP-1
```

To set dynamic weight by using the command line interface

At the command prompt, type:

set gslb vserver <name> -dynamicWeight DynamicWeightType

Example

```
set gslb vserver Vserver-GSLB-1 -dynamicWeight ServiceWeight
```

To add weights to the GSLB services by using the command line interface

At the command prompt, type:

set gslb vserver <name> -serviceName GSLBServiceName -weight WeightValue

Example

```
set gslb vserver Vserver-GSLB-1 -serviceName Service-GSLB-1 -weight 1
```

To configure a virtual server to assign weights to services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. Select the virtual server (for example, Vserver-LB-1) and click Open.
3. On the Services tab, in the Weights spin box, type or select the weight of a service (for example, 4) next to Service-HTTP-1).
4. Click OK.

To add weights to the GSLB services by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. Select the virtual server (for example vserver-GSLB-1) and click Open.

3. On the Services tab, in the Weight spin box, type or select the weight of a service (for example, next to service-GSLB-1, type 1).
4. Click OK.

To set dynamic weight by using the configuration utility

1. Navigate to Traffic Management > GSLB > Virtual Servers.
2. Select the virtual server (for example vserver-GSLB-1) and click Open.
3. On the Method and Persistence tab, under Method, in Dynamic Weight drop-down list, select SERVICEWEIGHT.
4. Click OK.

Configuring for Disaster Recovery with Data Center Persistence

Updated: 2014-11-24

Data center persistence is required for web applications that require maintaining a connection with the same server instead of having the requests load balanced. For example, in an e-commerce portal, maintaining a connection between the client and the same server is critical. For such applications, HTTP redirect persistence can be configured in an active-active setup.

To configure GSLB for disaster recovery with data center persistence, you must first configure the basic GSLB set up and then configure HTTP redirect persistence.

First create two GSLB sites, local and remote. Next, for the local site, create a GSLB virtual server and GSLB services and bind the services to the virtual server. Next, create ADNS services and bind the domain for which you are configuring GSLB to the GSLB virtual server at the local site. Next, create a load balancing virtual server with the same virtual server IP address as the GSLB service. Finally, duplicate the previous steps for the remote configuration, or configure the NetScaler appliance to autosynchronize your GSLB configuration.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

Once you have configured a basic GSLB setup, configure HTTP redirect precedence to enable data center persistence.

To configure HTTP redirect by using the command line interface

At the command prompt, type the following commands to configure HTTP redirect and verify the configuration:

- set gslb service <serviceName> -sitePersistence <sitePersistence> -sitePrefix <string>
- show gslb service <serviceName>

Example

```
set gslb service Service-GSLB-1 -sitePersistence HTTPRedirect -sitePrefix vserver-GSLB-1
show gslb service Service-GSLB-1
```

To configure HTTP redirect by using the configuration utility

1. Navigate to Traffic Management > GSLB > Services.
2. Select the GSLB service to be configured and click Open.
3. On the Advanced tab, under Site Persistence options, select the HTTPRedirect option.
4. In the Site Prefix text box, enter the site prefix (for example, vserver-GSLB-1).
5. Click OK.

Configuring GSLB for Proximity

Aug 30, 2013

When you configure GSLB for proximity, client requests are forwarded to the closest data center. The main benefit of the proximity-based GSLB method is faster response times resulting from the selection of the closest available data center. Such a deployment is critical for applications that require fast access to large volumes of data.

You can configure GSLB for proximity based on the round trip time (RTT), static proximity, or a combination of the two.

Configuring Dynamic Method (RTT)

Dynamic round trip time (RTT) is a measure of time or delay in the network between the client's local DNS server and a data resource. To measure dynamic RTT, the NetScaler appliance probes the client's local DNS server and gathers RTT metric information. The NetScaler then uses this metric to make its load balancing decision. Global server load balancing monitors the real-time status of the network and dynamically directs the client request to the data center with the lowest RTT value

To configure GSLB for proximity with dynamic method, you must first configure the basic GSLB set up and then configure dynamic RTT.

First create two GSLB sites, local and remote. Then, for the local site, create a GSLB virtual server and GSLB services and bind the services to the virtual server. Then create ADNS services and bind the domain for which you are configuring GSLB to the GSLB virtual server at the local site. Finally, create a load balancing virtual server with the same virtual server IP address as the GSLB service.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

Once you have configured a basic GSLB setup, configure the dynamic RTT method.

For details on how to configure the GSLB virtual server to use the dynamic RTT method for load balancing, see Configuring Dynamic RTT.

Configuring Static Proximity

The static proximity method for GSLB uses an IP address-based static proximity database to determine the proximity between the client's local DNS server and the GSLB sites. The NetScaler appliance responds with the IP address of a site that best matches the proximity criteria.

If two or more GSLB sites at different geographic locations serve the same content, the NetScaler appliance maintains a database of IP address ranges and uses the database for decisions about the GSLB sites to which to direct incoming client requests.

To configure GSLB for proximity with static proximity, you must first configure the basic GSLB set up and then configure static proximity.

First create two GSLB sites, local and remote. Then, for the local site, create a GSLB virtual server and GSLB services and bind the services to the virtual server. Then create ADNS services and bind the domain for which you are configuring GSLB to the GSLB virtual server at the local site. Finally, create a load balancing virtual server with the same virtual server IP address as the GSLB service.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

Once you have configured a basic GSLB setup, configure static proximity.

For details on how to configure the GSLB virtual server to use static proximity for load balancing, see Configuring Static Proximity.

Configuring Static Proximity and Dynamic RTT

You can configure the GSLB virtual server to use a combination of static proximity and dynamic RTT when you have some clients coming from an internal network like a branch office. You can configure GSLB such that the clients coming from the branch office or any other internal network are directed to a particular GSLB site that is geographically close to the client network. For all other requests, you can use dynamic RTT.

First create two GSLB sites, local and remote. Then, for the local site, create a GSLB virtual server and GSLB services and bind the services to the virtual server. Then create ADNS services and bind the domain for which you are configuring GSLB to the GSLB virtual server at the local site. Finally, create a load balancing virtual server with the same virtual server IP address as the GSLB service.

For details on how to configure a basic GSLB setup, see Configuring Global Server Load Balancing (GSLB).

Once you have configured a basic GSLB setup, configure the GSLB virtual server to use static proximity for all traffic originating from an internal network and then use dynamic RTT for all other traffic.

For details on how to configure static proximity, see Configuring Static Proximity and for details on how to configure dynamic RTT, see Configuring Dynamic RTT.

Configuring Parent-Child Topology

Aug 26, 2016

NetScaler appliances configured for global server load balancing (GSLB) provide for disaster recovery and ensure continuous availability of applications by protecting against points of failure in a wide area network (WAN). GSLB can balance the load across data centers by directing client requests to the closest or best performing data center, or to surviving data centers in the event of an outage.

There are three fundamental entities that must be configured for GSLB:

- **Site**: A GSLB site represents a NetScaler or a high availability (HA) pair of NetScaler appliances that maintain GSLB state information and provide information about how the NetScaler nodes should communicate. A site can also represent a data center.
- **GSLB virtual server**: A GSLB virtual server represents a group of resources to which users can be directed, and the logic used to select one resource verses another.
- **GSLB service**: A GSLB service represents a target resource and is bound to a GSLB virtual server. The target resource might be a load balancing virtual server on a NetScaler, or it could represent a third party server.

Sites and services are inherently linked to indicate proximity between the two. That is, all services must belong to a site, and are assumed to be in the same location as the GSLB site for proximity purposes. Likewise, services and virtual servers are linked, so that the logic is linked to the resources that are available.

Relationships among GSLB Sites

The concept of sites is central to NetScaler GSLB implementations. Unless otherwise specified, sites form a peer relationship among themselves. This relationship is used first to exchange health information and then to distribute load as determined by the selected algorithm. In many situations, however, a peer relationship among all GSLB sites is not desirable. Reasons for not having an all-peer implementation could be

1. To clearly separate GSLB sites. For example, to separate sites that participate in resolving DNS queries from the traffic management sites.
2. To reduce the volume of Metric Exchange Protocol (MEP) traffic, which increases exponentially with an increasing number of peer sites.

These goals can be achieved by using parent and child GSLB sites. Parent-child relationships can be used to build a two-level hierarchical GSLB design with the following characteristics:

- At the top level are parent sites that have peer relationships with other parents.
- Each parent can have multiple children, but each child can have only one parent.
- Each parent site exchanges health information with its children and with other parent sites.
- A child communicates only with its parent.

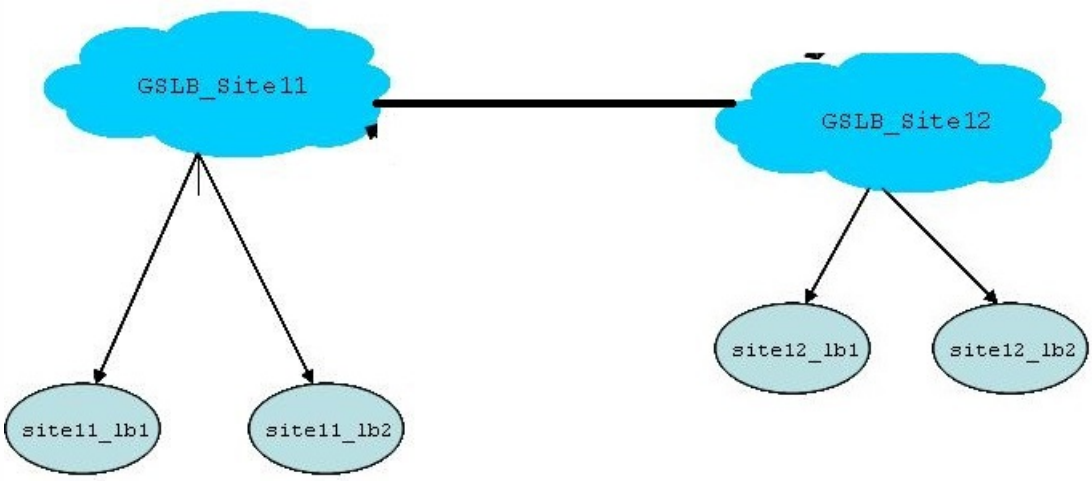
Note: In a parent-child relationship for GSLB, only the parent site does the GSLB resolution. The child sites act as normal load balancing sites.

Limitations of GSLB Parent-Child site configuration:

- You can configure 32 Parent sites and 1024 Child sites for each Parent site.
- On the Child site, by default, the nwmetricExchange and sessionExchange options are disabled.
- Round Trip Time (RTT) GSLB method is not recommended for GSLB Parent-Child site configuration.
- ADNS service or DNS load balancing virtual servers should be configured only in the Parent site.

Setting Up a Parent-Child Configuration for Global Server Load Balancing

If you have a firewall configured at a GSLB site, make sure that port 3011 is open. Follow the procedures at the following location to create services and virtual servers: Configuring Global Server Load Balancing (GSLB)
Figure 1. GSLB Parent-Child Topology



In the above figure:

- GSLB_Site11 and GSLB_Site12 are parent sites in a peer relationship.
- site11_lb1 and site11_lb2 are the child sites of GSLB_Site11, while site12_lb1 and site_lb2 are the child sites of GSLB_Site12.

The configuration of each parent site includes the information about all the child sites associated with it, but the configuration of each child site pertains only to that child and its parent. A child site is not aware about any other parent site or other child sites in the configuration. For example, in the above figure, the configuration of child site site11_lb1 would include only information about its parent site, GSLB_Site11.
Note: GSLB auto sync syncs only the GSLB configuration across the parent sites. It does not sync any configuration to the child sites.

To set up a parent-child configuration for GSLB by using the NetScaler command line

1. On each parent site, enter the following command: add gslb site<siteName><siteIPAddress> [-publicIP<ip_addr|ipv6_addr|*>] [-parentSite<string>] For example: # add gslb site gslb_site11 1.1.1.1 -publicIP 1.1.1.1
   # add gslb site site11_lb1 1.1.1.2 -publicIP 1.1.1.2 -parentSite gslb_site11

   # add gslb site site11_lb2 1.1.1.3 -publicIP 1.1.1.3 -parentSite gslb_site11

   # add gslb site gslb_site12 3.3.3.1 -publicIP 3.3.3.1

   # add gslb site site12_lb1 3.3.3.2 -publicIP 3.3.3.2 -parentSite gslb_site12

   # add gslb site site12_lb2 3.3.3.3 -publicIP 3.3.3.3 -parentSite gslb_site12

   The above command makes the parent site aware of its child sites as well as of the other parent site in the configuration.

2. On each child site, enter the following command: add gslb site<siteName><siteIPAddress> [-publicIP<ip_addr|ipv6_addr|*>] [-parentSite<string>] For example: # add gslb site site11_lb1 1 1.1.1.1 -publicIP

1.1.1.1

```
# add gslb site site11_lb2 1 1.1.1.2 -publicIP 1.1.1.2 -parentSite gslb_site11
```

The above command creates the child site and adds the parent-site information to child site's configuration.

Network metrics, such as RTT and persistence session information, are synced only across the parent sites. Therefore, parameters like nwMetric and sessionExchange are disabled by default on all the child sites.

To verify correct parent-child configuration, check the states of all the GSLB services bound to the parent sites.

Note: If you want to use different private and public IP address for GSLB services, add the corresponding GSLB-service related configuration to the child site in a separate procedure, not as part of the GSLB site configuration.

Link Load Balancing

May 19, 2015

Link load balancing (LLB) balances outbound traffic across multiple Internet connections provided by different service providers. LLB enables the Citrix® NetScaler® appliance to monitor and control traffic so that packets are transmitted seamlessly over the best possible link. Unlike with server load balancing, where a service represents a server, with LLB, a service represents a router or the next hop. A link is a connection between the NetScaler and the router.

To configure link load balancing, many users begin by configuring a basic setup with default settings. Configuring a basic setup involves configuring services, virtual servers, monitors, routes, an LLB method, and, optionally, configuring persistence. Once a basic setup is operational, you can customize it for your environment.

Load balancing methods that are applicable to LLB are round robin, destination IP hash, least bandwidth, and least packets. You can optionally configure persistence for connections to be sustained on a specific link. The available persistence types are source IP address-based, destination IP address-based, and source IP and destination IP address-based. PING is the default monitor but configuring a transparent monitor is recommended.

You can customize your setup by configuring reverse NAT (RNAT) and backup links.

This document includes the following information:

- Configuring a Basic LLB Setup
- Configuring RNAT with LLB
- Configuring a Backup Route
- Resilient LLB Deployment Scenario
- Monitoring an LLB Setup

Configuring a Basic LLB Setup

May 19, 2015

To configure LLB, you first create services representing each router to the Internet Service Providers (ISPs). A PING monitor is bound by default to each service. Binding a transparent monitor is optional but recommended. Then, you create a virtual server, bind the services to the virtual server, and configure a route for the virtual server. The route identifies the virtual server as the gateway to the physical routers represented by the services. The virtual server selects a router by using the load balancing method that you specify. Optionally, you can configure persistence to make sure that all traffic for a particular session is sent over a specific link.

To configure a basic LLB setup, do the following:

- Configure services
- Configure an LLB virtual server and binding a service
- Configure the LLB method and persistence
- Configure an LLB route
- Create and bind a transparent monitor

Configuring Services

Updated: 2014-10-27

A default monitor (PING) is automatically bound to a service type of ANY when the service is created, but you can replace the default monitor with a transparent monitor, as described in "Creating and Binding a Transparent Monitor."

To create a service by using the command line interface

At the command prompt, type:

- add service <name> <IP> <serviceType> <port>
- show service <name>

Example

```
add service ISP1R_svc_any 10.10.10.254 any *
show service ISP1R_svc_any
      ISP1R_svc_any (10.10.10.254:*) - ANY
      State: DOWN
      Last state change was at Tue Aug 31 04:31:13 2010
      Time since last state change: 2 days, 05:34:18.600
      Server Name: 10.10.10.254
      Server ID : 0   Monitor Threshold : 0
      Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
      Use Source IP: NO
      Client Keepalive(CKA): NO
      Access Down Service: NO
      TCP Buffering(TCPB): YES
```

```
    HTTP Compression(CMP): NO
    Idle timeout: Client: 120 sec  Server: 120 sec
    Client IP: DISABLED
    Cacheable: NO
    SC: OFF
    SP: OFF
    Down state flush: ENABLED

1)  Monitor Name: ping
        State: UP      Weight: 1
        Probes: 244705  Failed [Total: 0 Current: 0]
        Last response: Success - ICMP echo reply received.
        Response Time: 1.322 millisec
 Done
```

To create services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters:

   - Service Name*—name
   - Server—IP
   - Protocol*—serviceType (Select ANY from the drop-down list.)
   - Port*—port

   * A required parameter

4. Click Create.
5. Repeat Steps 2-4 to create another service.
6. Click Close.
7. In the Services pane, select the services that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configuring an LLB Virtual Server and Binding a Service

Updated: 2014-10-28

After you create a service, create a virtual server and bind services to the virtual server. The default LB method of least connections is not supported in LLB. For information about changing the LB method, see "Configuring the LLB Method and Persistence."

To create a link load balancing virtual server and bind a service by using the command line interface

At the command prompt, type:

- add lb vserver <name> <serviceType>
- bind lb vserver < name> <serviceName>
- show lb vserver < name>

Example


```
add lb vserver Router1-vip any
bind lb vserver Router-vip ISP1R_svc_any
sh lb vserver router-vip
        Router-vip (0.0.0.0:0) - ANY    Type: ADDRESS
        State: DOWN
        Last state change was at Thu Sep  2 10:51:32 2010
        Time since last state change: 0 days, 17:51:46.770
        Effective State: DOWN
        Client Idle Timeout: 120 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        No. of Bound Services :  1 (Total)      0 (Active)
        Configured Method: ROUNDROBIN
        Mode: IP
        Persistence: NONE
        Connection Failover: DISABLED

1) ISP1R_svc_any (10.10.10.254: *) - ANY State: DOWN    Weight: 1
 Done
```

To create a link load balancing virtual server and bind a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers pane, click Add.
3. In the Create Virtual Servers (Load Balancing) dialog box, specify values for the following parameters:

   - Name*—name
   - Protocol*—serviceType (Select ANY.)

   * A required parameter

   Note: Make sure Directly Addressable is unchecked.
4. Under the Services tab, in the Active column, select the check box for the service that you want to bind to the virtual server.
5. Click Create, and then click Close.
6. In the Load Balancing Virtual Servers tab, select the virtual server that you just created, and verify that the settings displayed in the Details pane are correct.

Configuring the LLB Method and Persistence

Updated: 2014-10-28

By default, the NetScaler appliance uses the least connections method to select the service for redirecting each client request, but you should set the LLB method to one of the supported methods. You can also configure persistence, so that different transmissions from the same client are directed to the same server.

To configure the LLB method and/or persistence by using the command line interface

At the command prompt, type the following command:

- set lb vserver <name> -lbMethod <lbMethod> -persistenceType <persistenceType>
- show lb vserver <name>

Example

```
set lb vserver router-vip -lbmethod ROUNDROBIN -persistencetype SOURCEIP

show lb vserver Router-vip
        Router-vip (0.0.0.0:0) - ANY    Type: ADDRESS
        State: DOWN
        Last state change was at Fri Sep  3 04:46:48 2010
        Time since last state change: 0 days, 00:52:21.200
        Effective State: DOWN
        Client Idle Timeout: 120 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        No. of Bound Services :  0 (Total)      0 (Active)
        Configured Method: ROUNDROBIN
        Mode: IP
        Persistence: SOURCEIP
     Persistence Mask: 255.255.255.255      Persistence v6MaskLength: 128   Persistence Timeout: 2 min
        Connection Failover: DISABLED
```

To configure the link load balancing method and/or persistence by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the Load Balancing method and/or persistence settings, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Method and Persistence tab, specify values for some or all of the following parameters:
   - Method—lbMethod
   - Persistence—persistenceType
4. Click OK.
5. In the Load Balancing Virtual Servers pane, select the virtual server that you just configured and verify that the settings displayed in the Details pane are correct.

Configuring an LLB Route

Updated: 2014-10-28

After configuring the IPv4 or IPv6 services, virtual servers, LLB methods, and persistence, you configure an IPv4 or IPv6 LLB route for the network specifying the virtual server as the gateway. A route is a collection of links that are load balanced. Requests are sent to the virtual server IP address that acts as the gateway for all outbound traffic and selects the router based on the LLB method configured.

To configure an IPv4 LLB route by using the command line interface

At the command prompt, type:

- add lb route <network> <netmask> <gatewayName>
- show lb route [<network> <netmask>]

Example

```
add lb route 0.0.0.0 0.0.0.0 Router-vip
show lb route 0.0.0.0 0.0.0.0
     Network         Netmask       Gateway/VIP          Flags
    -----------     -------------  --------------        --------
1)    0.0.0.0         0.0.0.0       Router-vip            UP
```

To configure an IPv6 LLB route by using the command line interface

At the command prompt, type:

- add lb route6 <network> <gatewayName>
- show lb route6

```
add lb route6 ::/0 llb6_vs
show lb route6
     Network          VIP           Flags
    -----------      ---------       --------
1)    ::/0            llb6_vs           UP
```

Example

To configure an LLB route by using the configuration utility

1. Navigate to System > Network > Routes.
2. In the details pane, select one of the following:
   - Click LLB to configure an IPv4 route.
   - Click LLBV6 to configure an IPv4 route.
3. In the Create LB Route or Create LB IPV6 Routedialog box, set the following parameters:

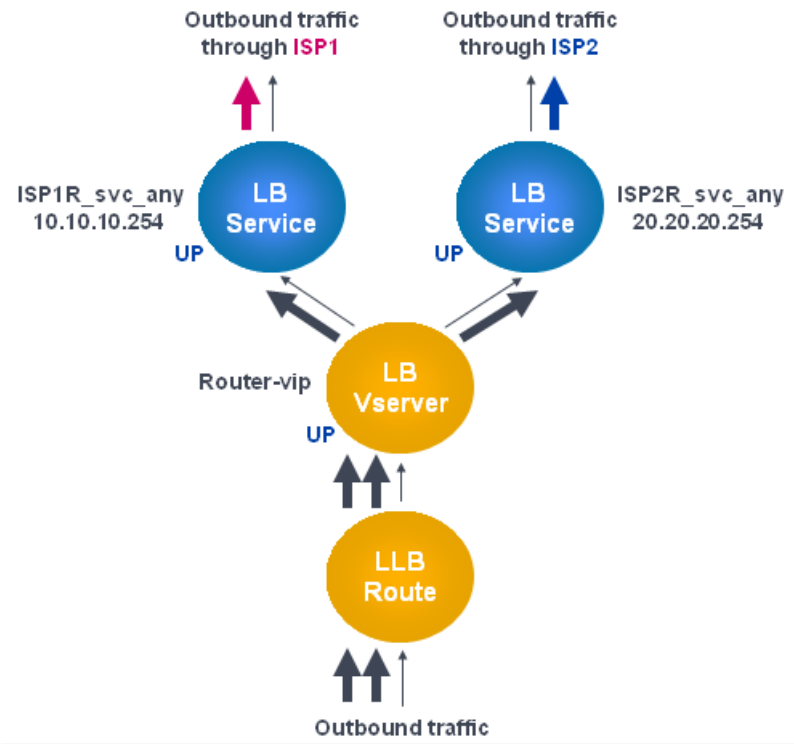
   - Network*
   - Netmask*—Required for IPV4 routes.
   - Gateway Name*—gatewayName

   * A required parameter

4. Click Create, and then click Close. The route that you just created appears on the LLB or the LLB6 tab in the Routes pane.

The following diagram shows a basic LLB setup. A service is configured for each of the two links (ISPs) and PING monitors are bound by default to these services. A link is selected based on the LLB method configured.

Figure 1. Basic LLB Setup



Note: If your Internet service provider has provided an IPv6 address, replace the IPv4 service with an IPv6 service in the above figure.

Creating and Binding a Transparent Monitor

Updated: 2014-10-28

You create a transparent monitor to monitor the health of upstream devices, such as routers. You can then bind the transparent monitor to services. The default PING monitor monitors the connectivity only between the NetScaler appliance and the upstream device. The transparent monitor monitors all the devices existing in the path from the appliance to the device that owns the destination IP address specified in the monitor. If a transparent monitor is not configured and the status of the router is UP but one of the next hop devices from that router is down, the appliance includes the router while performing load balancing and forwards the packet to the router. However, the packet is not delivered to the final destination because one of the next hop devices is down. By binding a transparent monitor, if any of the devices (including the router) are down, the service is marked as DOWN and the router is not included when the appliance performs link load balancing.

To create a transparent monitor by using the command line interface

At the command prompt, type:

- add lb monitor <monitorName> <type> -destIP <ip_addr|*> -transparent YES
- show lb monitor [<monitorName>]

Example

```
add lb monitor monitor-1 PING -destIP 10.10.10.11 -transparent YES
> show lb monitor monitor-1
1)  Name.....: monitor-1  Type.....:    PING   State....:   ENABLED
Standard parameters:
```

```
Interval.....:          5 sec   Retries.....:            3
Response timeout.:        2 sec   Down time.....:        30 sec
Reverse.....:           NO   Transparent.....:          YES
Secure.....:            NO   LRTM.....:        ENABLED
Action.....:   Not applicable   Deviation.....:         0 sec
Destination IP...:     10.10.10.11
Destination port.:    Bound service
Iptunnel.....:          NO
TOS.....:           NO   TOS ID.....:            0
SNMP Alert Retries:           0     Success Retries..:           1
Failure Retries..:          0
```

To create a transparent monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the Monitors pane, click Add.
3. In the Create Monitor dialog box, set the following parameters:

   - Name*
   - Type*
   - Destination IP
   - Transparent

   * A required parameter

4. Click Create, and then click Close.
5. In the Monitors pane, select the monitor that you just configured and verify that the settings displayed in the Details pane are correct.

To bind a monitor to a service by using the command line interface

At the command prompt, type:

- bind lb monitor <monitorName> <serviceName>
- show service <name>

Example

```
bind lb monitor monitor-HTTP-1 isp1R_svc_any
 Done
> show service isP1R_svc_any
      ISP1R_svc_any (10.10.10.254:*) - ANY
      State: UP
      Last state change was at Thu Sep  2 10:51:07 2010
      Time since last state change: 0 days, 18:41:55.130
      Server Name: 10.10.10.254
      Server ID : 0   Monitor Threshold : 0
      Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
      Use Source IP: NO
      Client Keepalive(CKA): NO
```

```
    Access Down Service: NO
    TCP Buffering(TCPB): YES
    HTTP Compression(CMP): NO
    Idle timeout: Client: 120 sec  Server: 120 sec
    Client IP: DISABLED
    Cacheable: NO
    SC: OFF
    SP: OFF
    Down state flush: ENABLED

1)  Monitor Name: monitor-HTTP-1
        State: UP  Weight: 1
        Probes: 1256  Failed [Total: 0 Current: 0]
        Last response: Success - ICMP echo reply received.
        Response Time: 1.322 millisec
Done
```

To bind a monitor to a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select a service to which you want to bind a monitor, and then click Open.
3. In the Configure Service dialog box, on the Monitors tab, under Available, select the monitor that you want to bind to the service, and then click Add.
4. Click OK.
5. In the Services pane, select the service that you just configured and verify that the settings displayed in the Details pane are correct.

Configuring RNAT with LLB

Oct 28, 2014

You can configure an LLB setup for reverse network address translation (RNAT) for outbound traffic. This ensures that the return network traffic for a specific flow is routed through the same path. First configure basic LLB, as described in "Configuring a Basic LLB Setup", and then configure RNAT. You must then enable use subnet IP (USNIP) mode.

To configure RNAT by using the command line interface

At the command prompt, type:

- set rnat <network> <netmask>
- show rnat

Example

```
set rnat 10.102.29.0 255.255.255.0
> show rnat
1)     Network: 10.102.29.0    Netmask: 255.255.255.0
       NatIP: *
```

To configure RNAT by using the configuration utility

1. Navigate to System > Network > Routes.
2. In the detailspane, on the RNAT tab, click Configure RNAT.
3. In the Configure RNAT dialog box, specify values for the following parameters:
   - Network*—network
   - Netmask*—netmask

   * A required parameter
4. Click Create, and then click Close. The RNAT route that you just created appears in the on the RNAT tab in the Routes pane.

To enable Use Subnet IP mode by using the command line interface

At the command prompt, type:

- enable ns mode USNIP
- show ns mode

Example

```
enable ns mode USNIP
> show ns mode
      Mode                  Acronym          Status
      -------               -------          ------
 1)   Fast Ramp                FR              ON
 2)   ....
 8)   Use Subnet IP             USNIP            ON
```

9) ...

To enable Use Subnet IP mode by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In details pane, under Modes and Features, click Configure modes.
3. In the Configure Modes dialog box, select Use Subnet IP, and then click OK.
4. In the Enable/Disable Mode(s) message box, click Yes.

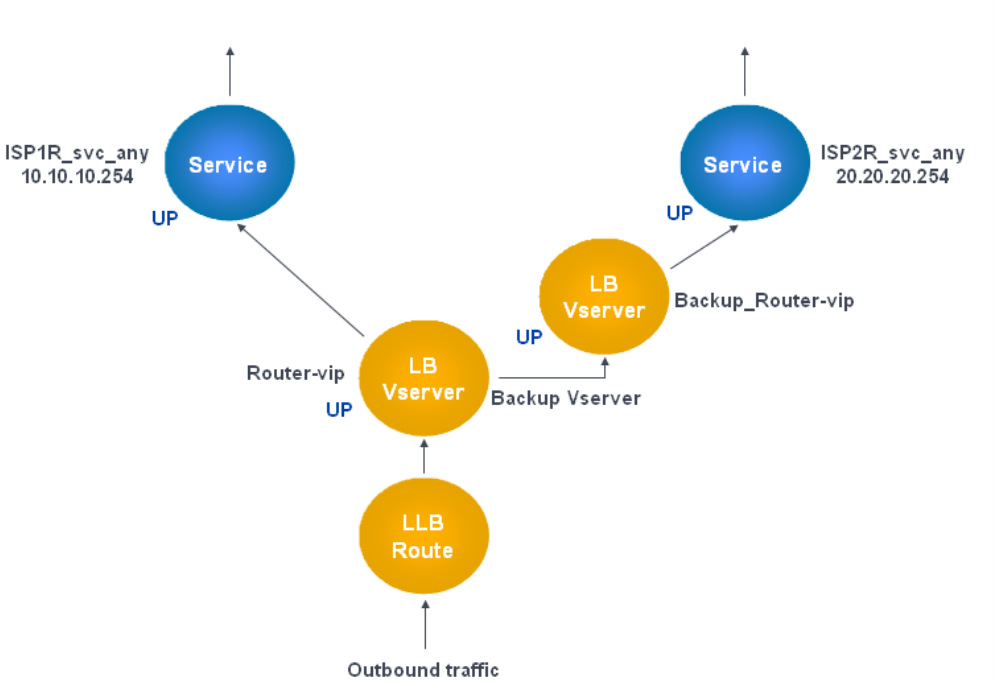
Configuring a Backup Route

Oct 28, 2014

To prevent disruption in services when the primary route is down, you can configure a backup route. Once the backup route is configured, the NetScaler appliance automatically uses it when the primary route fails. First create a primary virtual server as described in "Configuring an LLB Virtual Server and Binding a Service." To configure a backup route, create a secondary virtual server similar to a primary virtual server and then designate this virtual server as a backup virtual server (route).

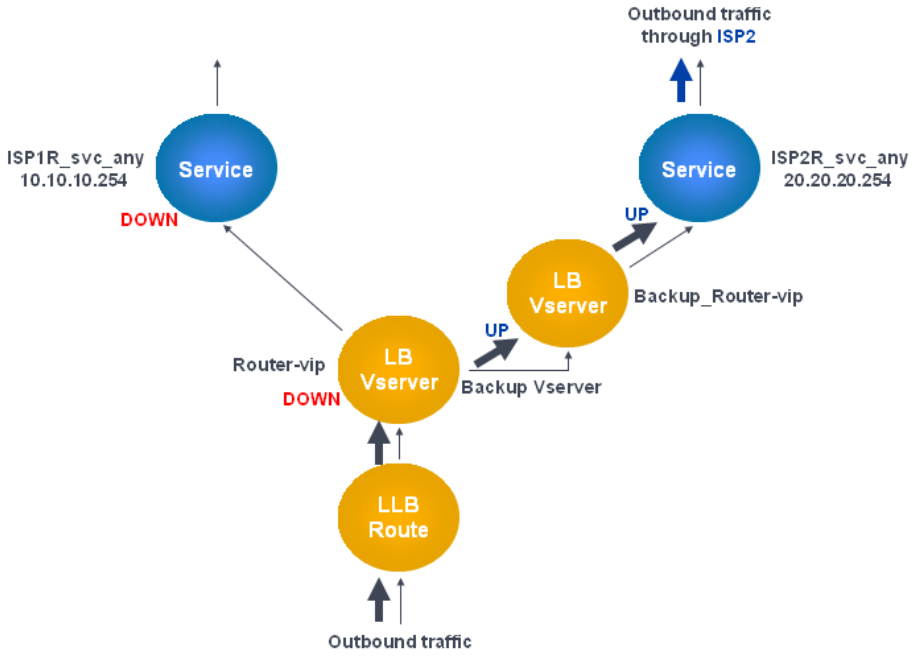
In the following diagram, **Router-vip** is the primary virtual server, and **Backup_Router-vip** is the secondary virtual server designated as the backup virtual server.

Figure 1. Backup Route Setup



Note: If your Internet service provider has provided an IPv6 address, replace the IPv4 service with an IPv6 service in the above figure.
By default, all traffic is sent through the primary route. However, when the primary route fails, all traffic is diverted to the backup route as shown in the following diagram.

Figure 2. Backup Routing in Operation



Note: If your Internet service provider has provided an IPv6 address, replace the IPv4 service with an IPv6 service in the above figure.

To set the secondary virtual server as the backup virtual server by using the command line interface

At the command prompt, type:

set lb vserver <name> -backupVserver <string>

Example

```
set lb vserver Router-vip -backupVServer Backup_Router-vip
> show lb vserver Router-vip
      Router-vip (0.0.0.0:0) - ANY    Type: ADDRESS
      State: UP
      Last state change was at Fri Sep  3 04:46:48 2010
      Time since last state change: 0 days, 03:09:45.600
      Effective State: UP
      Client Idle Timeout: 120 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      No. of Bound Services :  1 (Total)      1 (Active)
      Configured Method: ROUNDROBIN
      Mode: IP
      Persistence: DESTIP     Persistence Mask: 255.255.255.255      Persistence v6MaskLength: 128   Persistence Timeout: 2 min
      Backup: Router2-vip
      Connection Failover: DISABLED
 Done
```

To set the secondary virtual server as the backup virtual server by using the configuration utility

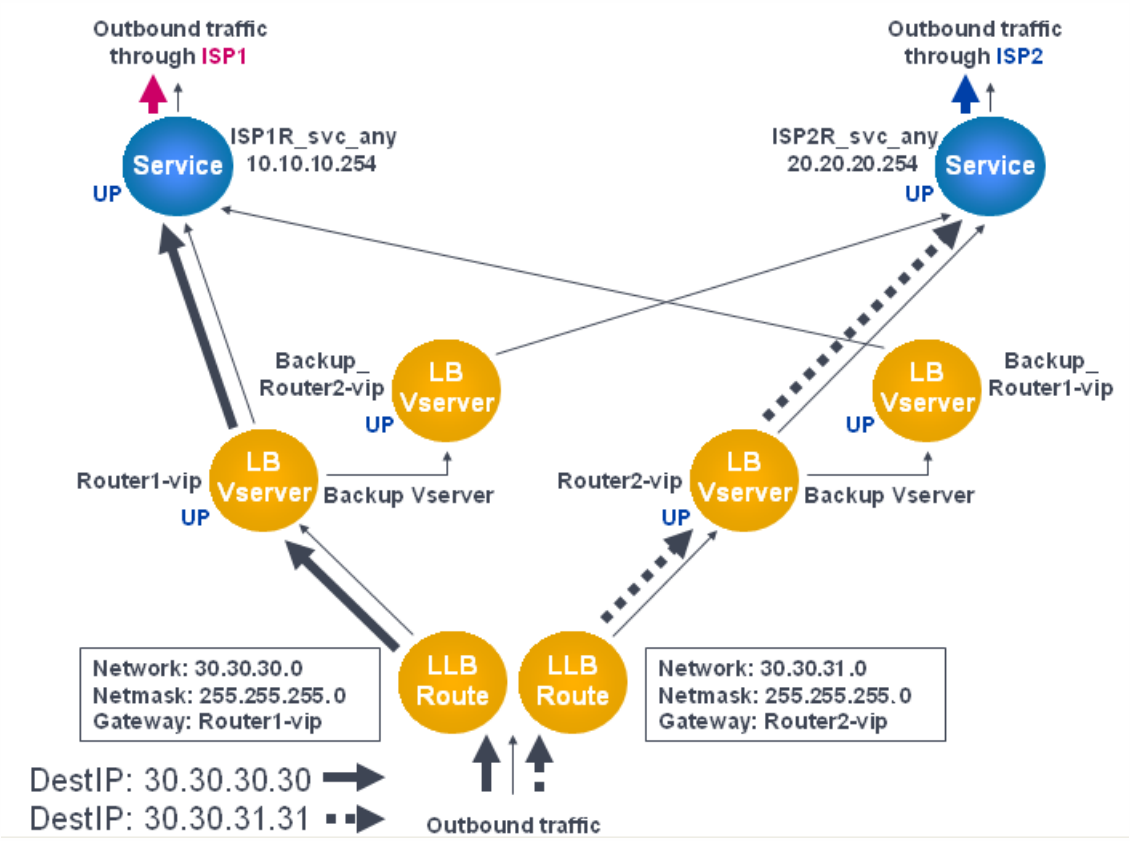
1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the secondary virtual server for which you want to configure the backup virtual server, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. In the Backup Virtual Server drop-down list, select the secondary backup virtual server, and then click OK.

Resilient LLB Deployment Scenario

Mar 22, 2012

In the following diagram, there are two networks: 30.30.30.0 and 30.30.31.0. Link load balancing is configured based on the destination IP address. Two routes are configured with gateways **Router1-vip** and **Router2-vip**, respectively. **Router1-vip** is configured as a backup to **Router2-vip** and vice versa. All traffic with the destination IP specified as 30.30.30.30 is sent through **Router1-vip** and traffic with the destination IP specified as 30.30.31.31 is sent through **Router2-vip**.

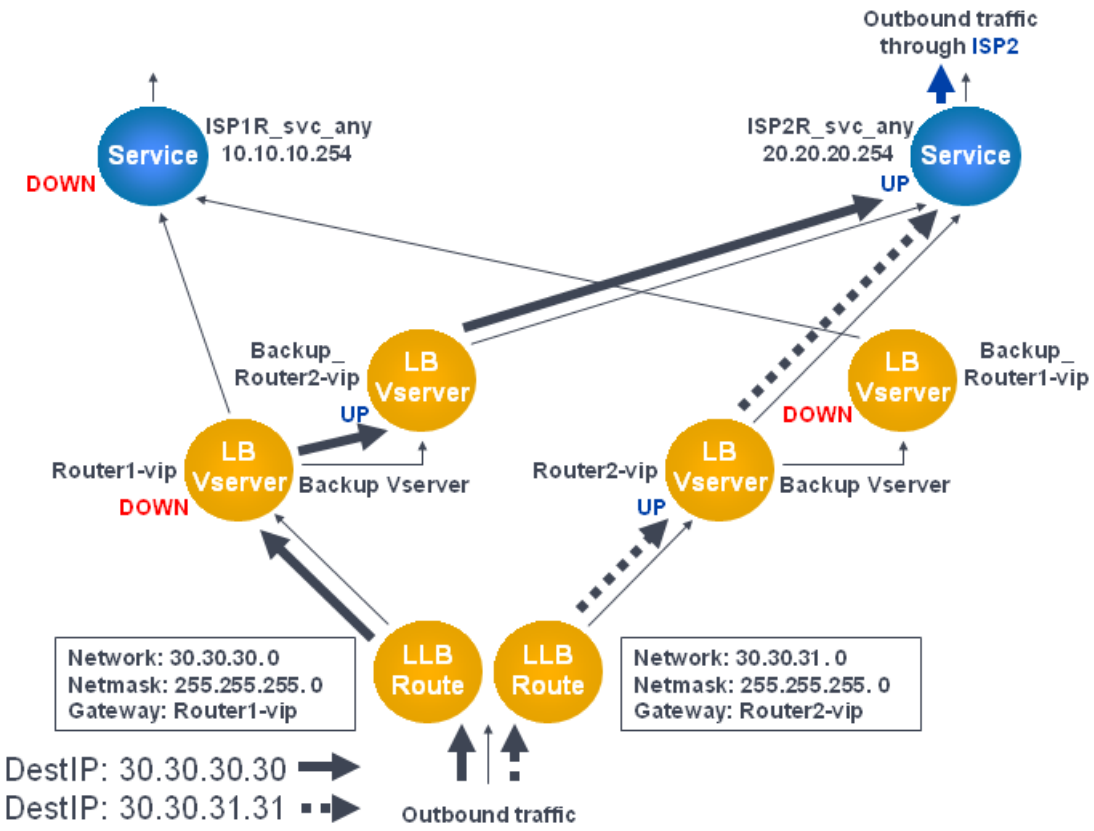
Figure 1. Resilient LLB Deployment Setup



Note: If your Internet service provider has provided an IPv6 address, replace the IPv4 service with an IPv6 service in the above figure.

However, if any one of the gateways (**Router1-vip** or **Router2-vip**) is DOWN, traffic is routed through the backup router. In the following diagram, **Router1-vip** for ISP1 is DOWN, so all traffic with the destination IP specified as 30.30.30.30 is also sent through ISP2.

Figure 2. Resilient LLB Deployment Scenario



Note: If your Internet service provider has provided an IPv6 address, replace the IPv4 service with an IPv6 service in the above figure.

Monitoring an LLB Setup

Sep 20, 2010

After the configuration is up and running, you should view the statistics for each service and virtual server to check for possible problems.

Viewing the Statistics of a Virtual Server

Updated: 2013-09-05

To evaluate the performance of virtual servers or to troubleshoot problems, you can display details of the virtual servers configured on the NetScaler appliance. You can display a summary of statistics for all the virtual servers, or you can specify the name of a virtual server to display the statistics only for that virtual server. You can display the following details:

- Name
- IP address
- Port
- Protocol
- State of the virtual server
- Rate of requests received
- Rate of hits

To display virtual server statistics by using the command line interface

To display a summary of the statistics for all the virtual servers currently configured on the NetScaler, or for a single virtual server, at the command prompt, type:

stat lb vserver [-detail] [<name>]

Example

```
>stat lb vserver -detail
Virtual Server(s) Summary
                vsvrIP  port    Protocol      State   Req/s   Hits/s
One                  *    80        HTTP         UP     5/s     0/s
Two                  *     0         TCP       DOWN     0/s     0/s
Three                *  2598         TCP       DOWN     0/s     0/s
dnsVirtualNS    10.102.29.90    53         DNS       DOWN     0/s     0/s
BRVSERV          10.10.1.1    80        HTTP       DOWN     0/s     0/s
LBVIP         10.102.29.66    80        HTTP         UP     0/s     0/s
 Done
```

To display virtual server statistics by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. If you want to display the statistics for only one virtual server, in the details pane, select the virtual server whose statistics you want to display.
3. In the details pane, click Statistics.

Viewing the Statistics of a Service

Updated: 2013-08-28

You can view the rate of requests, responses, request bytes, response bytes, current client connections, requests in surge queue, current server connections, and so forth using the service statistics.

To view the statistics of a service by using the command line interface

At the command prompt, type:

stat service <name>
Example

stat service Service-HTTP-1

To view the statistics of a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service whose statistics you want to view (for example, Service-HTTP-1).
3. Click Statistics. The statistics appear in a new window.

Load Balancing

Mar 24, 2015

The load balancing feature distributes user requests for web pages and other protected applications across multiple servers that all host (or mirror) the same content. You use load balancing primarily to manage user requests to heavily used applications, preventing poor performance and outages and ensuring that users can access your protected applications. Load balancing also provides fault tolerance; when one server that hosts a protected application becomes unavailable, the feature distributes user requests to the other servers that host the same application.

You can configure the load balancing feature to:

- Distribute all requests for a specific protected website, application, or resource between two or more identically configured servers.
- Use any of several different algorithms to determine which server should receive each incoming user request, basing the decision on different factors, such as which server has the fewest current user connections or which server has the lightest load.

The load balancing feature is a core feature of the NetScaler appliance. Most users first set up a working basic configuration and then customize various settings, including persistence for connections. In addition, you can configure features for protecting the configuration against failure, managing client traffic, managing and monitoring servers, and managing a large scale deployment.

How Load Balancing Works

Jun 08, 2015

In a basic load balancing setup, clients send their requests to the IP address of a virtual server configured on the NetScaler appliance. The virtual server distributes them to the load-balanced application servers according to a preset pattern, called the load balancing algorithm. In some cases, you might want to assign the load balancing virtual server a wildcard address instead of a specific IP address. For instructions about specifying a global HTTP port on the appliance, see Global HTTP Ports.

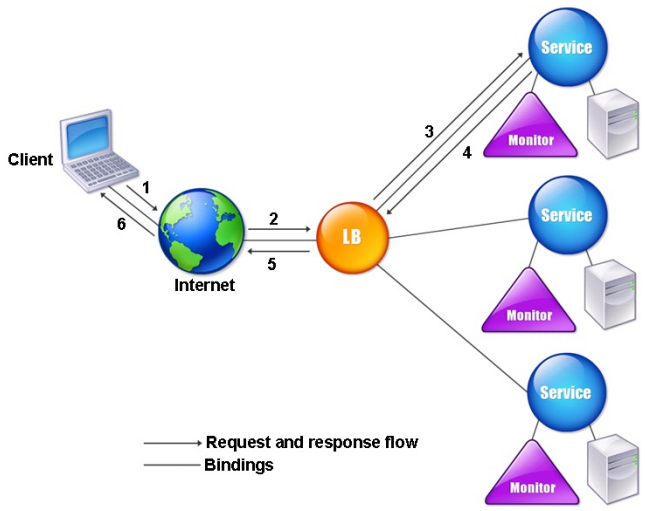
To understand how load balancing works, see the following sections:

- Load Balancing Basics
- Understanding the Topology
- Use of Wildcards Instead of IP Addresses and Ports
- Configuring Global HTTP Ports

Load Balancing Basics

A load balancing setup includes a load-balancing virtual server and multiple load-balanced application servers. The virtual server receives incoming client requests, uses the load balancing algorithm to select an application server, and forwards the requests to the selected application server. The following conceptual drawing illustrates a typical load balancing deployment. Another variation involves assigning a global HTTP port.

Figure 1. Load Balancing Architecture



The load balancing virtual server can use any of a number of algorithms (or methods) to determine how to distribute load among the load-balanced servers that it manages. The default load balancing method is the least connection method, in which the NetScaler appliance forwards each incoming client connection to whichever load-balanced application server currently has the fewest active user connections.

The entities that you configure in a typical NetScaler load balancing setup are:

- **Load balancing virtual server.** The IP address, port, and protocol combination to which a client sends connection

requests for a particular load-balanced website or application. If the application is accessible from the Internet, the virtual server IP (VIP) address is a public IP address. If the application is accessible only from the local area network (LAN) or wide area network (WAN), the VIP is usually a private (ICANN non-routable) IP address.

- **Service.** The IP address, port, and protocol combination used to route requests to a specific load-balanced application server. A service can be a logical representation of the application server itself, or of an application running on a server that hosts multiple applications. After creating a service, you bind it to a load balancing virtual server.
- **Server object.** A virtual entity that enables you to assign a name to a physical server instead of identifying the server by its IP address. If you create a server object, you can specify its name instead of the server's IP address when you create a service. Otherwise, you must specify the server's IP address when you create a service, and the IP address becomes the name of the server.
- **Monitor.** An entity on the NetScaler appliance that tracks a service and ensures that it is operating correctly. The monitor periodically probes (or performs a health check on) each service to which you assign it. If the service does not respond within the time specified by the time-out, and a specified number of health checks fail, that service is marked DOWN. The NetScaler appliance then skips that service when performing load balancing, until the issues that caused the service to quit responding are fixed.

The virtual server, services, and load balanced application servers in a load balancing setup can use either Internet Protocol version 4 (IPv4) or Internet Protocol version 6 (IPv6) IP addresses. You can mix IPv4 and IPv6 addresses in a single load balancing setup.

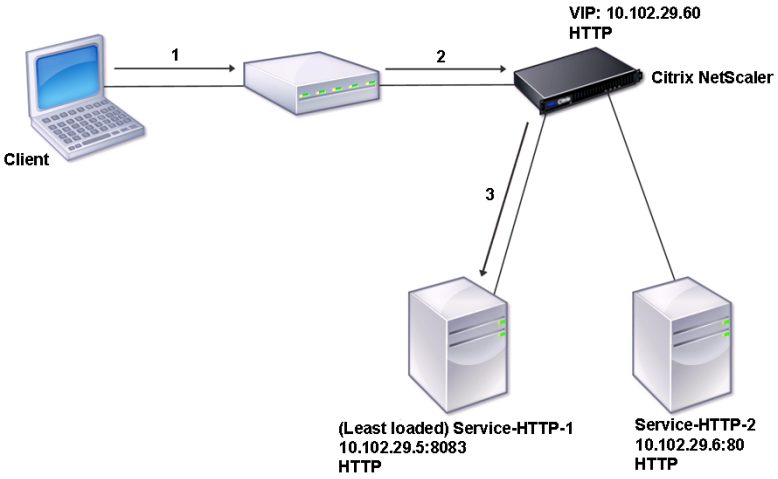
For variations in the load balancing setup, see the following use cases:

- Configuring Load Balancing in Direct Server Return Mode
- Configuring LINUX Servers in DSR Mode
- Configuring DSR Mode When Using TOS
- Configuring Load Balancing in DSR Mode by Using IP Over IP
- Configuring Load Balancing in One-arm Mode
- Configuring Load Balancing in the Inline Mode
- Load Balancing of Intrusion Detection System Servers
- Load Balancing RDP services

Understanding the Topology

In a load balancing setup, the load balancing server is logically located between the client and the server farm, and manages traffic flow to the servers in the server farm. On the NetScaler appliance, the application servers are represented by virtual entities called services. The following diagram shows the topology of a basic load balancing configuration.

Figure 2. Basic Load Balancing Topology

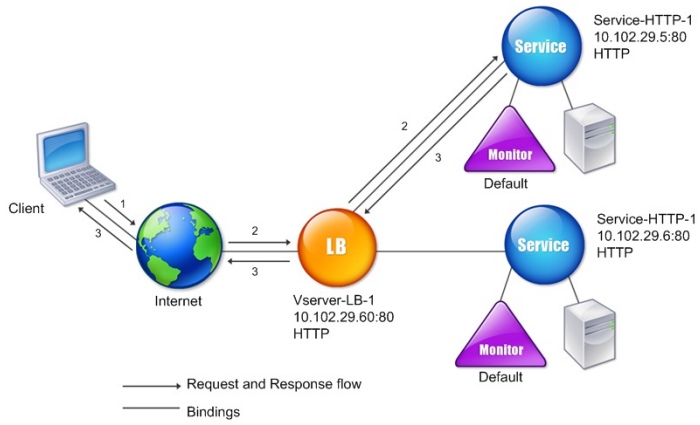


In the diagram, load balancing is used to manage traffic flow to the servers. The virtual server selects the service and assigns it to serve client requests. Consider a scenario where the services Service-HTTP-1 and Service-HTTP-2 are created and bound to the virtual server named Vserver-LB-1. Vserver-LB-1 forwards the client request to either Service-HTTP-1 or Service-HTTP-2. The NetScaler appliance uses the least connection load balancing method to select the service for each request. The following table lists the names and values of the basic entities that must be configured on the appliance.

Entity	**Mandatory Parameters and Sample Values**			
	Name	**IP Address**	**Port**	**Protocol**
Virtual server	Vserver-LB-1	10.102.29.60	80	HTTP
Services	Service-HTTP-1	10.102.29.5	8083	HTTP
	Service-HTTP-2	10.102.29.6	80	HTTP
Monitors	Default	None	None	None

The following diagram shows the load balancing sample values and mandatory parameters that are described in the preceding table.

Figure 3. Load Balancing Entity Model



Use of Wildcards Instead of IP Addresses and Ports

Updated: 2013-11-20

In some cases you might need to use a wildcard for the IP address or the port of a virtual server or for the port of a service. The following cases may require using a wildcard:

- If the NetScaler appliance is configured as a transparent pass through, which must accept all traffic that is sent to it regardless of the IP or port to which it is sent.
- If one or more services listen on ports that are not well known.
- If one or more services, over time, change the ports that they listen on.
- If you reach the limit for the number of IP addresses and ports that you can configure on a single NetScaler appliance.
- If you want to create virtual servers that listen for all traffic on a specific virtual LAN.

When a wildcard-configured virtual server or service receives traffic, the NetScaler appliance determines the actual IP address or port and creates new records for the service and associated load balanced application server. These dynamically created records are called dynamically learned server and service records.

For example, a firewall load balancing configuration can use wildcards for both the IP address and port. If you bind a wildcard TCP service to this type of load balancing virtual server, the virtual server receives and processes all TCP traffic that does not match any other service or virtual server.

The following table describes some of the different types of wildcard configurations and when each should be used.

IP	Port	Protocol	Description
*	*	TCP	A general wildcard virtual server that accepts traffic sent to any IP address and port on the NetScaler appliance. When using a wildcarded virtual server, the appliance dynamically learns the IP and port of each service and creates the necessary records as it processes traffic.
*	*	TCP	A firewall load balancing virtual server. You can bind firewall services to this virtual server,

IP	Port	Protocol	Description
IP Address	*	TCP,UDP, and ANY	A virtual server that accepts all traffic that is sent to the specified IP address, regardless of the port. You must explicitly bind to this type of virtual server the services to which it will redirect traffic. It will not dynamically learn them. Note: You do not configure services or virtual servers for a global HTTP port. In this case, you configure a specific port as a global HTTP port (for example, set ns param -httpPort 80). The appliance then accepts all traffic that matches the port number, and processes it as HTTP traffic. The appliance dynamically learns and creates services for this traffic.
*	port	SSL, SSL_TCP	A virtual server that accepts all traffic sent to any IP address on a specific port. Used for global transparent SSL offloading. All SSL, HTTP, and TCP processing that usually is performed for a service of the same protocol type is applied to traffic that is directed to this specific port. The appliance uses the port to dynamically learn the IP of the service it should use. If –cleartext is not specified, the NetScaler appliance uses end-to-end SSL.
*	port	Not applicable	All other virtual servers that can accept traffic to the port. You do not bind services to these virtual servers; the NetScaler appliance learns them dynamically.

Note: If you have configured your NetScaler appliance as a transparent pass through that makes use of global (wildcard) ports, you may want to turn on Edge mode. For more information, see "Configuring Edge Mode."
The NetScaler appliance attempts to locate virtual servers and services by first attempting an exact match. If none is found, it continues to search for a match based on wildcards, in the following order:

1. Specific IP address and specific port number
2. Specific IP address and a * (wildcard) port
3. * (wildcard) IP address and a specific port
4. * (wildcard) IP address and a * (wildcard) port

If the appliance is unable to select a virtual server by IP address or port number, it searches for a virtual server on the basis of the protocol used in the request, in the following order:

1. HTTP
2. TCP
3. ANY

Configuring Global HTTP Ports

Updated: 2013-10-23

You do not configure services or virtual servers for a global HTTP port. Instead, you configure a specific port by using the set ns param command. After configuring this port, the NetScaler appliance accepts all traffic that matches the port number, and processes it as HTTP traffic, dynamically learning and creating services for that traffic.

You can configure more than one port number as a global HTTP port. If you are specifying more than one port number in a single set ns param command, separate the port numbers by a single white space. If one or more ports have already been

specified as global HTTP ports, and you want to add one or more ports without removing the ports that are currently configured, you must specify all the port numbers, current and new, in the command. Before you add port numbers, use the **show ns param** command to view the ports that are currently configured.

To configure a global HTTP port by using the command line interface

At the command prompt, type the following commands to configure a global HTTP port and verify the configuration:

- set ns param –httpPort <port>
- show ns param

Example 1: Configuring a port as a global HTTP port

In this example, port 80 is configured as a global HTTP port.

```
> set ns param -httpPort 80
 Done
> show ns param
        Global configuration settings:
                     HTTP port(s): 80
                  Max connections: 0
          Max requests per connection: 0
                Client IP insertion: DISABLED
                   Cookie version: 0
       Persistence Cookie Secure Flag: ENABLED
       ...
       ...
```

Example 2: Adding ports when one or more global HTTP ports are already configured

In this example, port 8888 is added to the global HTTP port list. Port 80 is already configured as a global HTTP port.

```
> show ns param

       Global configuration settings:
                     HTTP port(s): 80
                  Max connections: 0
          Max requests per connection: 0
                Client IP insertion: DISABLED
                   Cookie version: 0
       Persistence Cookie Secure Flag: ENABLED
                     Min Path MTU: 576
       ...
       ...
 Done
> set ns param -httpPort 80 8888
 Done
>  show ns param

       Global configuration settings:
                     HTTP port(s): 80,8888
```



```
                Max connections: 0
          Max requests per connection: 0
                Client IP insertion: DISABLED
                   Cookie version: 0
         Persistence Cookie Secure Flag: ENABLED
                   Min Path MTU: 576

        ...
        ...
 Done
>
```

To configure a global HTTP port by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Settings, click Change HTTP parameters.
3. In the Configure HTTP Parameters dialog box, in the HTTP Port area, do the following:
   - To add a port, enter the port number, and then click Add.
   - To remove a port that has already been configured, click the port number, and then click Remove.
4. Click OK.

Setting Up Basic Load Balancing

Oct 22, 2015

Before configuring your initial load balancing setup, enable the load balancing feature. Then begin by creating at least one service for each server in the load balancing group. With the services configured, you are ready to create a load balancing virtual server, and bind each service to the virtual server. That completes the initial setup. Before proceeding with further configuration, verify your configuration to make sure that each element was configured properly and is operating as expected.

To set up basic load balancing, see the following sections:

- Enabling Load Balancing
- Configuring Services
- Creating a Virtual Server
- Binding Services to the Virtual Server
- Verifying the Configuration

Enabling Load Balancing

You can configure load balancing entities such as services and virtual servers when the load balancing feature is disabled, but they will not function until you enable the feature.

To enable load balancing by using the command line interface

At the command prompt, type the following command to enable load balancing and verify the configuration:

- enable ns feature LB
- show ns feature

Example

```
> enable ns feature LoadBalancing
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection         SP              ON
 3)    Load Balancing           LB              ON
 .
 .
 .
 24)   NetScaler Push           push            OFF
 Done
```

To enable load balancing by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.

2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Load Balancing check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes.

Configuring Services

After you enable the load balancing feature, you must create at least one service for each application server that is to be included in your load balancing setup. The services that you configure provide the connections between the NetScaler appliance and the load balanced servers. Each service has a name and specifies an IP address, a port, and the type of data that is served.

If you create a service without first creating a server object, the IP address of the service is also the name of the server that hosts the service. If you prefer to identify servers by name rather than IP address, you can create server objects and then specify a server's name instead of its IP address when you create a service.

When you create a service that uses UDP as the transport layer protocol, a ping monitor is automatically bound to the service. A ping monitor is the most basic of the built-in monitors. When you create a service that uses TCP as the transport layer protocol, a TCP_default monitor is automatically bound to the service. When you develop a strategy for managing your load balancing setup, you might decide to bind a different type of monitor, or multiple monitors, to the service.

Creating a Service

Before you create a service, you need to understand the different service types and how each is used. The following list describes the types of services supported on the NetScaler appliance.

HTTP

Used for load-balanced servers that accept HTTP traffic, such as standard web sites and web applications. The HTTP service type enables the NetScaler appliance to provide compression, content filtering, caching, and client keep-alive support for your Layer 7 web servers. This service type also supports virtual server IP port insertion, redirect port rewriting, Web 2.0 Push, and URL redirection support.

Because HTTP is a TCP-based application protocol, you can also use the TCP service type for web servers. If you do so, however, the NetScaler appliance is able to perform only Layer 4 load balancing. It cannot provide any of the Layer 7 support described earlier.

SSL

Used for servers that accept HTTPS traffic, such as ecommerce web sites and shopping cart applications. The SSL service type enables the NetScaler appliance to encrypt and decrypt SSL traffic (perform SSL offloading) for your secure web applications. It also supports HTTP persistence, content switching, rewrite, virtual server IP port insertion, Web 2.0 Push, and URL redirection.

You can also use the SSL_BRIDGE, SSL_TCP, or TCP service types. If you do so, however, the NetScaler performs only Layer 4 load balancing. It cannot provide SSL offloading or any of the Layer 7 support described above

FTP

Used for servers that accept FTP traffic. The FTP service type enables the NetScaler appliance to support specific details of the FTP protocol.

You can also use TCP or ANY service types for FTP servers.

TCP

Used for servers that accept many different types of TCP traffic, or that accept a type of TCP traffic for which a more specific type of service is not available.

You can also use the ANY service type for these servers.

SSL_TCP

Used for servers that accept non-HTTP-based SSL traffic, to support SSL offloading.

You can also use the TCP service type for these services. If you do so, the NetScaler appliance performs both the Layer 4 load balancing and SSL offloading.

UDP

Used for servers that accept UDP traffic. You can also use the ANY service type.

SSL_BRIDGE

Used for servers that accept SSL traffic when you do not want the NetScaler appliance to perform SSL offloading. Alternatively, you can use the SSL_TCP service type.

NNTP

Used for servers that accept Network News Transfer Protocol (NNTP) traffic, typically Usenet sites.

DNS

Used for servers that accept DNS traffic, typically nameservers. With the DNS service type, the NetScaler appliance validates the packet format of each DNS request and response. It can also cache DNS responses. You can apply DNS policies to DNS services.

You can also use the UDP service type for these services. If you do, however, the NetScaler appliance can only perform Layer 4 load balancing. It cannot provide support for DNS-specific features.

ANY

Used for servers that accept any type of TCP, UDP, or ICMP traffic. The ANY parameter is used primarily with firewall load balancing and link load balancing.

SIP-UDP

Used for servers that accept UDP-based Session Initiation Protocol (SIP) traffic. SIP initiates, manages, and terminates multimedia communications sessions, and has emerged as the standard for Internet telephony (VoIP).

You can also use the UDP service type for these services. If you do, however, the NetScaler appliance performs only Layer 4 load balancing. It cannot provide support for SIP-specific features.

DNS-TCP

Used for servers that accept DNS traffic, where the NetScaler appliance acts as a proxy for TCP traffic sent to DNS servers. With services of the DNS-TCP service type, the NetScaler appliance validates the packet format of each DNS request and response and can cache DNS responses, just as with the DNS service type.

You can also use the TCP service type for these services. If you do, however, the NetScaler appliance only performs Layer 4 load balancing of external DNS name servers. It cannot provide support for any DNS-specific features.

RTSP

Used for servers that accept Real Time Streaming Protocol (RTSP) traffic. RTSP provides delivery of multimedia and other streaming data. Select this type to support audio, video, and other types of streamed media.

You can also use the TCP service type for these services. If you do, however, the NetScaler appliance performs only Layer 4 load balancing. It cannot parse the RTSP stream or provide support for RTSPID persistence or RTSP NATting.

DHCPRA

Used for servers that accept DHCP traffic. The DHCPRA service type can be used to relay DHCP requests and responses between VLANs.

DIAMETER

Used for load balancing Diameter traffic among multiple Diameter servers. Diameter uses message-based load balancing.

SSL_DIAMETER

Used for load balancing Diameter traffic over SSL.

Services are designated as DISABLED until the NetScaler appliance connects to the associated load-balanced server and verifies that it is operational. At that point, the service is designated as ENABLED. Thereafter, the NetScaler appliance periodically monitors the status of the servers, and places any that fail to respond to monitoring probes (called health checks) back in the DISABLED state until they respond.

Note: You can create a range of services from a single CLI command or the same dialog box. The names in the range vary by a number used as a suffix/prefix. For example, service1, service2, and so on. From the configuration utility, you can specify a range only in the last octet of the IP address, which is the fourth in case of an IPv4 address and the eighth in case of an IPv6 address. From the command line, you can specify the range in any octet of the IP address.

To create a service by using the command line interface

At the command prompt, type:

add service <name> <serverName> <serviceType> <port>

Example COPY

```
add service Service-HTTP-1 192.0.2.5 HTTP 80
```

To create a service by using the configuration utility

1. Navigate to **Traffic Management** > **Load Balancing** > **Services**.
2. In the details pane, click **Add**.
3. In the **Create Service** dialog box, specify values for the following parameters:

- Service Name—name
- Server—serverName
- Protocol—serviceType
- Port—port

4. Click **Create**, and then click **Close**. The service you created appears in the **Services** pane.

Creating a Virtual Server

After you create your services, you must create a virtual server to accept traffic for the load balanced Web sites, applications, or servers. Once load balancing is configured, users connect to the load-balanced Web site, application, or server through the virtual server's IP address or FQDN.

Note: The virtual server is designated as DOWN until you bind the services that you created to it, and until the NetScaler appliance connects to those services and verifies that they are operational. Only then is the virtual server designated as UP.

To create a virtual server by using the command line interface

At the command prompt, type:

add lb vserver <name> <serviceType> <ip> <port>
Example

```
add lb vserver Vserver-LB-1 HTTP 10.102.29.60 80
```

To create a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters:
   - Name—name
   - IP Address—IPAddress
     Note: If the virtual server uses IPv6, select the IPv6 check box and enter the address in IPv6 format (for example, 1000:0000:0000:0000:0005:0600:700a:888b).
   - Protocol—serviceType
   - Port—port
4. Click Create, and then click Close. The virtual server you created appears in the Load Balancing Virtual Servers pane.

Binding Services to the Virtual Server

After you have created services and a virtual server, you must bind the services to the virtual server. In most cases, services are bound to virtual servers of the same type, but you can bind certain types of services to certain different types of virtual servers, as shown below.

Virtual Server Type	Service Type	Comment
HTTP	SSL	You would normally bind an SSL service to an HTTP virtual server to do encryption.
SSL	HTTP	You would normally bind an HTTP service to an SSL virtual server to do SSL offloading.

Virtual Server Type	Service Type	Comment
SSL_TCP	TCP	You would normally bind a TCP service to an SSL_TCP virtual server to do SSL offloading for other TCP (SSL decryption without content awareness).

The state of the services bound to a virtual server determines the state of the virtual server: if all of the bound services are DOWN, the virtual server is marked DOWN, and if any of the bound services is UP or OUT OF SERVICE, the state of the virtual server is UP.

To bind a service to a load balancing virtual server by using the command line interface

At the command prompt, type:

bind lb vserver <name> <serviceName>
Example

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

To bind a service to a load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to bind the service.
3. Click Open.
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Services tab, select the Active check box next to the service that you want to bind to the virtual server.
5. Click OK.

Note: You can bind a service to multiple virtual servers.

Verifying the Configuration

After finishing your basic configuration, you should view the properties of each service and load balancing virtual server in your load balancing setup to verify that each is configured correctly. After the configuration is up and running, you should view the statistics for each service and load balancing virtual server to check for possible problems.

Viewing the Properties of a Server Object

You can view properties such as the name, state, and IP address of any server object in your NetScaler appliance configuration.

To view the properties of server objects by using the command line interface

At the command prompt, type:

show server <serverName>

Example COPY

```
show server server-1
```

To view the properties of server objects by using the configuration utility

Navigate to **Traffic Management** > **Load Balancing** > **Servers**. The parameter values of the available servers appear in the details pane.

Viewing the Properties of a Virtual Server

You can view properties such as the name, state, effective state, IP address, port, protocol, method, and number of bound services for your virtual servers. If you have configured more than the basic load balancing settings, you can view the persistence settings for your virtual servers, any policies that are bound to them, and any cache redirection and content switching virtual servers that have been bound to the virtual servers.

To view the properties of a load balancing virtual server by using the command line interface

At the command prompt, type:

show lb vserver <name>

Example COPY

```
show lb vserver Vserver-LB-1
```

To view the properties of a load balancing virtual server by using the configuration utility

1. Navigate to **Traffic Management** > **Load Balancing** > **Virtual Servers**.

2. In the details pane, click a virtual server to display its properties at the bottom of the

details pane.

3. To view cache redirection and content switching virtual servers that are bound to this

virtual server, click **Show CS/CR Bindings**.

Viewing the Properties of a Service

You can view the name, state, IP address, port, protocol, maximum client connection, maximum requests per connection, and server type of the configured services, and use this information to troubleshoot any mistake in the service configuration.

To view the properties of services by using the command line interface

At the command prompt, type:

show service <name>

Example COPY

```
show service Service-HTTP-1
```

To view the properties of services by using the configuration utility

Navigate to **Traffic Management** > **Load Balancing** > **Services**. The details of the available services appear on the **Services** pane.

Viewing the Bindings of a Service

You can view the list of virtual servers to which the service is bound. The binding information also provides the name, IP address, port and state of the virtual servers to which the services are bound. You can use the binding information to troubleshoot any problem with binding the services to virtual servers.

To view the bindings of a service by using the command line

At the command prompt, type:

show service bindings <name>

Example COPY

```
show service bindings Service-HTTP-1
```

To view the bindings of a service by using the configuration utility

1. Navigate to **Traffic Management** > **Load Balancing** > **Services**.

2. In the details pane, select the service whose binding information you want to view.

3. In the **Action** tab, click **Show Bindings**.

Viewing the Statistics of a Virtual Server

To evaluate the performance of virtual servers or to troubleshoot problems, you can display details of the virtual servers configured on the NetScaler appliance. You can display a summary of statistics for all the virtual servers, or you can specify the name of a virtual server to display the statistics only for that virtual server. You can display the following details:

- Name

- IP address
- Port
- Protocol
- State of the virtual server
- Rate of requests received
- Rate of hits

To display virtual server statistics by using the command line interface

To display a summary of the statistics for all the virtual servers currently configured on the appliance, or for a single virtual server, at the command prompt, type:

stat lb vserver [-detail] [<name>]

Example COPY

```
>stat lb vserver -detail

Virtual Server(s) Summary

vsvrIP port Protocol State Req/s Hits/s

One * 80 HTTP UP 5/s 0/s

Two * 0 TCP DOWN 0/s 0/s

Three * 2598 TCP DOWN 0/s 0/s

dnsVirtualNS 10.102.29.90 53 DNS DOWN 0/s 0/s

BRVSERV 10.10.1.1 80 HTTP DOWN 0/s 0/s

LBVIP 10.102.29.66 80 HTTP UP 0/s 0/s

Done
```

To display virtual server statistics by using the configuration utility

1. Navigate to **Traffic Management** > **Load Balancing** > **Virtual Servers**.

2. If you want to display the statistics for only one virtual server, in the details pane, select the virtual server whose statistics you want to display.

3. In the details pane, click **Statistics**.

Viewing the Statistics of a Service

You can view the rate of requests, responses, request bytes, response bytes, current client connections, requests in surge queue, current server connections, and so forth using the service statistics.

To view the statistics of a service by using the command line interface

At the command prompt, type:

stat service <name>

Example COPY

```
stat service Service-HTTP-1
```

To view the statistics of a service by using the configuration utility

1. Navigate to **Traffic Management** > **Load Balancing** > **Services**.

2. In the details pane, select the service whose statistics you want to view (for example, Service-HTTP-1).

3. Click **Statistics**. The statistics appear in a new window.

Load Balancing Algorithms

Aug 29, 2013

The load balancing algorithm defines the criteria that the NetScaler appliance uses to select the service to which to redirect each client request. Different load balancing algorithms use different criteria. For example, the least connection algorithm selects the service with the fewest active connections, while the round robin algorithm maintains a running queue of active services, distributes each connection to the next service in the queue, and then sends that service to the end of the queue.

Some load balancing algorithms are best suited to handling traffic on websites, others to managing traffic to DNS servers, and others to handling complex web applications used in e-commerce or on company LANs or WANs. The following table lists each load balancing algorithm that the NetScaler appliance supports, with a brief description of how each operates.

Name	Server Selection Based On
LEASTCONNECTION	Which service currently has the fewest client connections. This is the default load balancing algorithm.
ROUNDROBIN	Which service is at the top of a list of services. After that service is selected for a connection, it moves to the bottom of the list.
LEASTRESPONSETIME	Which load balanced server currently has the quickest response time.
URLHASH	A hash of the destination URL.
DOMAINHASH	A hash of the destination domain.
DESTINATIONIPHASH	A hash of the destination IP address.
SOURCEIPHASH	A hash of the source IP address.
SRCIPDESTIPHASH	A hash of the source and destination IP addresses.
CALLIDHASH	A hash of the call ID in the SIP header.
SRCIPSRCPORTHASH	A hash of the client's IP address and port.
LEASTBANDWIDTH	Which service currently has the fewest bandwidth constraints.
LEASTPACKETS	Which service currently is receiving the fewest packets.

Name	Server Selection Based On
CUSTOMLOAD	Data from a load monitor.
TOKEN	The configured token.
LRTM	Fewest active connections and the lowest average response time.

Depending on the protocol of the service that it is load balancing, the NetScaler appliance sets up each connection between client and server to last for a different time interval. This is called load balancing granularity, of which are three types: request-based, connection-based, and time-based granularity. The following table describes each type of granularity and when each is used.

Granularity	Types of Load Balanced Service	Specifies
Request - based	HTTP or HTTPS	A new service is chosen for each HTTP request, independent of TCP connections. As with all HTTP requests, after the Web server fulfills the request, the connection is closed.
Connection-based	TCP and TCP-based protocols other than HTTP	A service is chosen for every new TCP connection. The connection persists until terminated by either the service or the client.
Time-based	UDP and other IP protocols	A new service is chosen for each UDP packet. Upon selection of a service, a session is created between the service and a client for a specified period of time. When the time expires, the session is deleted and a new service is chosen for any additional packets, even if those packets come from the same client.

During startup of a virtual server, or whenever the state of a virtual server changes, the virtual server can initially use the round robin method to distribute the client requests among the physical servers. This type of distribution, referred to as *startup round robin,* helps prevent unnecessary load on a single server as the initial requests are served. After using the round robin method at the startup, the virtual server switches to the load balancing method specified on the virtual server.

The Startup RR Factor works in the following manner:

- If the Startup RR Factor is set to zero, the NetScaler switches to the specified load balancing method depending on the request rate.
- If the Startup RR Factor is any number other than zero, NetScaler uses the round robin method for the specified number of requests before switching to the specified load balancing method.
- By default, the Startup RR Factor is set to zero.

Note: You cannot set the startup RR Factor for an individual virtual server. The value you specify applies to all the virtual servers on the NetScaler appliance.

To set the startup round-robin factor by using the command line interface

At the command prompt, type:

set lb parameter -startupRRFactor <positive_integer>

Example

```
set lb parameter -startupRRFactor 25000
```

To set the startup round-robin factor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. Under Settings, click Configure Load Balancing Parameters.
3. In the Configure Load Balancing Parameters dialog box, for Startup RR Factor type a value.
4. Click OK.

The Least Connection Method

Jul 10, 2013

When a virtual server is configured to use the least connection load balancing algorithm (or method), it selects the service with the fewest active connections. This is the default method, because, in most circumstances, it provides the best performance.

For TCP, HTTP, HTTPS, and SSL_TCP services, the NetScaler appliance includes the following connection types in its list of existing connections:

- **Active connections to a service.** Connections representing requests that a client has sent to the virtual server and that the virtual server has forwarded to a service. For HTTP and HTTPS services, active connections represent only those HTTP or HTTPS requests that have not yet received a response.
- **Waiting connections in the surge queue.** Any connections to the virtual server that are waiting in a surge queue and have not yet been forwarded to a service. Connections can build up in the surge queue at any time, for any of the following reasons:
  - Your services have connection limits, and all services in your load balancing configuration are at that limit.
  - The surge protection feature is configured and has been activated by a surge in requests to the virtual server.
  - The load-balanced server has reached an internal limit and therefore does not open any new connections. (For example, an Apache server's connection limit is reached.)

  When a virtual server uses the least connection method, it considers the waiting connections as belonging to the specific service. Therefore, it does not open new connections to those services.

For UDP services, the connections that the least connection algorithm considers include all sessions between the client and a service. These sessions are logical, time-based entities. When the first UDP packet in a session arrives, the NetScaler appliance creates a session between the source IP address and port and the destination IP address and port.

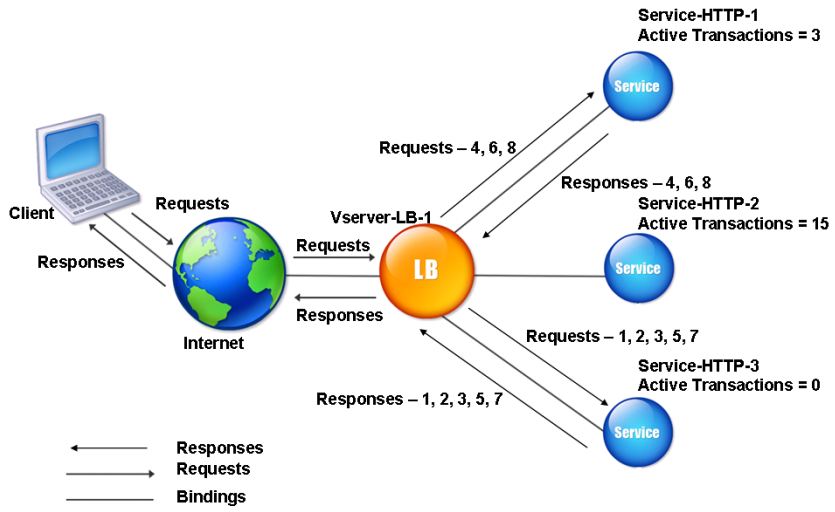
For Real-Time Streaming Protocol (RTSP) connections, the NetScaler appliance uses the number of active control connections to determine the lowest number of connections to an RTSP service.

The following example shows how a virtual server selects a service for load balancing by using the least connection method. Consider the following three services:

- Service-HTTP-1 is handling 3 active transactions.
- Service-HTTP-2 is handling 15 active transactions.
- Service-HTTP-3 is not handling any active transactions.

The following diagram illustrates how the NetScaler appliance forwards incoming requests when using the least connection method.

Figure 1. Mechanism of the Least Connections Load Balancing Method



In this diagram, the virtual server selects the service for each incoming connection by choosing the server with the fewest active transactions.

Connections are forwarded as follows:

- Service-HTTP-3 receives the first request, because it is not handling any active transactions.

  Note: The service with no active transaction is selected first.
- Service-HTTP-3 receives the second and third requests because the service has the next least number of active transactions.
- Service-HTTP-1 receives the fourth request Because Service-HTTP-1 and Service-HTTP-3 have same number of active transactions, the virtual server uses the round robin method to choose between them.
- Service-HTTP-3 receives the fifth request.
- Service-HTTP-1 receives the sixth request, and so on, until both Service-HTTP-1 and Service-HTTP-3 are handling the same number of requests as Service-HTTP-2. At that time, the NetScaler appliance starts forwarding requests to Service-HTTP-2 when it is the least loaded service or its turn comes up in the round robin queue.

  Note: If connections to Service-HTTP-2 close, it might get new connections before each of the other two services has 15 active transactions.

The following table explains how connections are distributed in the three-service load balancing setup described above.

Incoming Connection	Service Selected	Current Number of Active Connections	Remarks
Request-1	Service-HTTP-3 (N = 0)	1	Service-HTTP-3 has the fewest active connections.
Request-2	Service-	2	

Incoming Connection	Service Selected	Current Number of Active Connections	Remarks
	HTTP-3 (N = 1)		
Request-3	Service-HTTP-3 (N = 2)	3	
Request-4	Service-HTTP-1 (N = 3)	4	Service-HTTP-1 and Service-HTTP-3 have the same number of active connections.
Request-5	Service-HTTP-3 (N = 3)	4	
Request-6	Service-HTTP-1 (N = 4)	5	
Request-7	Service-HTTP-3 (N = 4)	5	
Request-8	Service-HTTP-1 (N = 5)	6	
Service-HTTP-2 is selected for load balancing when it completes its active transactions and the current connections to it close, or when the other services (Service-HTTP-1 and Service-HTTP-3) have 15 or more connections each.			

The NetScaler appliance can also use the least connection method when weights are assigned to services. It selects a service by using the value (Nw) of the following expression:

Nw = (Number of active transactions) * (10000 / weight)

The following example shows how the NetScaler appliance selects a service for load balancing by using the least connection method when weights are assigned to services. In the preceding example, suppose Service-HTTP-1 is assigned a weight of 2, Service-HTTP-2 is assigned a weight of 3, and Service-HTTP-3 is assigned a weight of 4. Connections are forwarded as follows:

- Service-HTTP-3 receives the first because the service is not handling any active transactions.

  Note: If services are not handling any active transactions, the NetScaler appliance uses the round robin method regardless of the weights assigned to each of the services.
- Service-HTTP-3 receives the second, third, fourth, fifth, sixth, and seventh requests because the service has lowest Nw value.
- Service-HTTP-1 receives the eighth request. Because Service-HTTP-1 and Service-HTTP-3 now have same Nw value, the NetScaler performs load balancing in a round robin manner. Therefore, Service-HTTP-3 receives the ninth request.

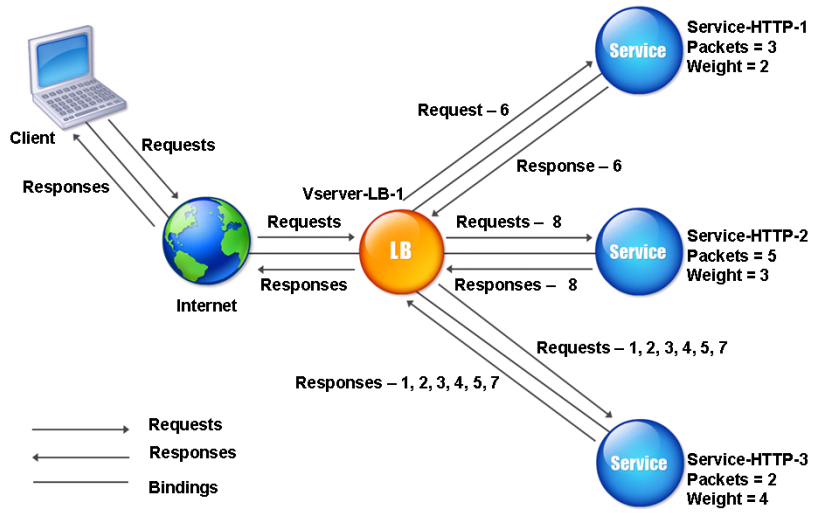
The following table explains how connections are distributed on the three-service load balancing setup that is described above.

Request Received	**Service Selected**	**Current Nw (Number of active transactions) * (10000 / weight) value**	**Remarks**
Request-1	Service-HTTP-3 (Nw = 0)	Nw = 2500	Service-HTTP-3 has the lowest Nw value.
Request-2	Service-HTTP-3 (Nw = 2500)	Nw = 5000	
Request-3	Service-HTTP-3 (Nw = 5000)	Nw = 7500	
Request-4	Service-HTTP-3 (Nw = 7500)	Nw = 10000	
Request-5	Service-HTTP-3 (Nw = 10000)	Nw = 12500	
Request-6	Service-HTTP-3	Nw = 15000	

Request Received	Service Selected	Current Nw (Number of active transactions) * (10000 / weight) value	Remarks
Request-7	Service-HTTP-1 (Nw = 15000)	Nw = 20000	Service-HTTP-1 and Service-HTTP-3 have the same Nw values
Request-8	Service-HTTP-3 (Nw = 15000)	Nw = 17500	
Service-HTTP-2 is selected for load balancing when it completes its active transactions or when the Nw value of other services (Service-HTTP-1 and Service-HTTP-3) is equal to 50000.			

The following diagram illustrates how the NetScaler appliance uses the least connection method when weights are assigned to the services.

Figure 2. Mechanism of the Least Connections Load Balancing Method when Weights are Assigned



To configure the least connection method, see Configuring a Load Balancing Method that Does Not Include a Policy.

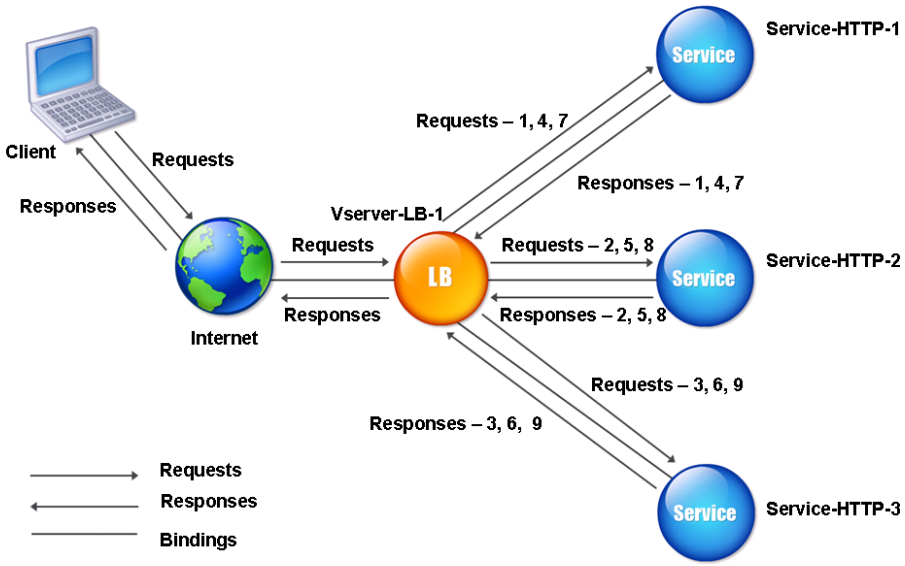
The Round Robin Method

Mar 16, 2012

When a load balancing virtual server is configured to use the round robin method, it continuously rotates a list of the services that are bound to it. When the virtual server receives a request, it assigns the connection to the first service in the list, and then moves that service to the bottom of the list.

The following diagram illustrates how the NetScaler appliance uses the round robin method with a load balancing setup that contains three load-balanced servers and their associated services.

Figure 1. How the Round Robin Load Balancing Method Works



If you assign a different weight to each service, the NetScaler appliance performs weighted round robin distribution of incoming connections. It does this by skipping the lower-weighted services at appropriate intervals.

For example, assume that you have a load balancing setup with three services. You set Service-HTTP-1 to a weight of 2, Service-HTTP-2 to a weight of 3, and Service-HTTP-3 to a weight of 4. The services are bound to Vserver-LB-1, which is configured to use the round robin method. With this setup, incoming requests are delivered as follows:

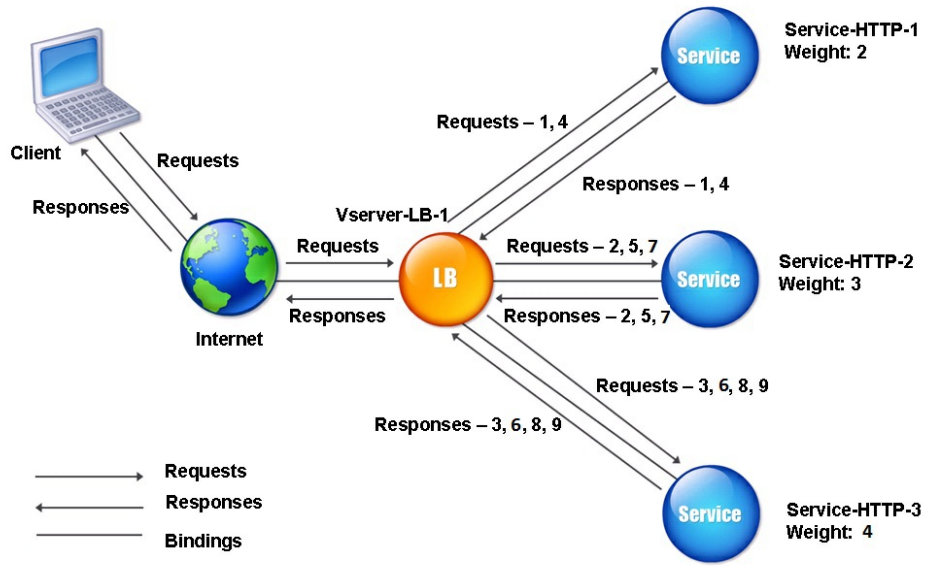
- Service-HTTP-1 receives the first request.
- Service-HTTP-2 receives the second request.
- Service-HTTP-3 receives the third request.
- Service-HTTP-1 receives the fourth request.
- Service-HTTP-2 receives the fifth request.
- Service-HTTP-3 receives the sixth request.
- Service-HTTP-2 receives the seventh request.
- Service-HTTP-3 receives both the eighth and the ninth requests.

Note: You can also configure weights on services to prevent multiple services from using the same server and overloading

the server.
A new cycle then begins, using the same pattern.

The following diagram illustrates the weighted round robin method.

Figure 2. How the Round Robin Load Balancing Method Works with Weighted Services



To configure the round robin method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Least Response Time Method

Jul 10, 2013

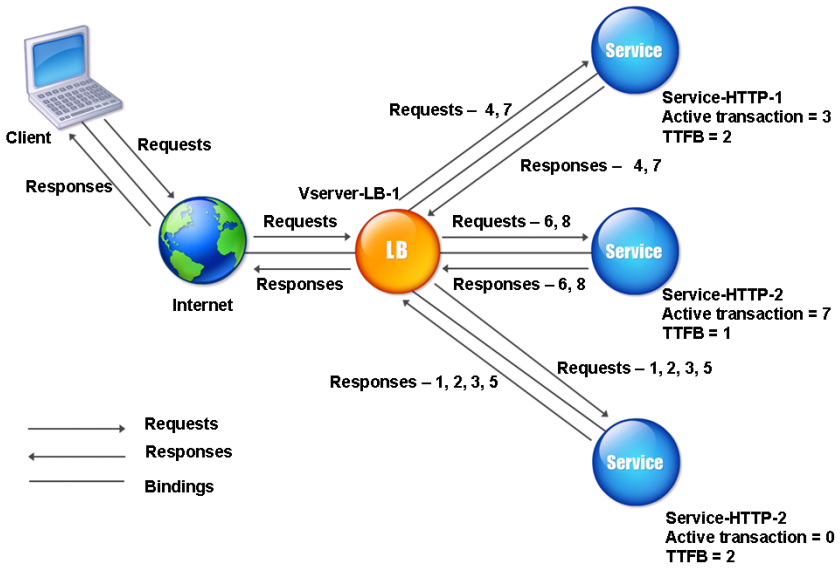
When the load balancing virtual server is configured to use the least response time method, it selects the service with the fewest active connections and the lowest average response time. You can configure this method for HTTP and Secure Sockets Layer (SSL) services only. The response time (also called Time to First Byte, or TTFB) is the time interval between sending a request packet to a service and receiving the first response packet from the service. The NetScaler appliance uses response code 200 to calculate TTFB.

The following example shows how a virtual server selects a service for load balancing by using the least response time method. Consider the following three services:

- Service-HTTP-1 is handling three active transactions and TTFB is two seconds.
- Service-HTTP-2 is handling seven active transactions and TTFB is one second.
- Service-HTTP-3 is not handling any active transactions and TTFB is two seconds.

The following diagram illustrates how the NetScaler appliance uses the least response time method to forward the connections.

Figure 1. How the Least Response Time Load Balancing Method Works



The virtual server selects a service by multiplying the number of active transactions by the TTFB for each service and then selecting the service with the lowest result. For the example shown above, the virtual server forwards requests as follows:

- Service-HTTP-3 receives the first request, because the service is not handling any active transactions.
- Service-HTTP-3 also receives the second and third requests, because the result is lowest of the three services.
- Service-HTTP-1 receives the fourth request. Since Service-HTTP-1 and Service-HTTP-3 have the same result, the NetScaler appliance chooses between them by applying the Round Robin method.
- Service-HTTP-3 receives the fifth request.
- Service-HTTP-2 receives the sixth request, because at this point it has the lowest result.
- Because Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 all have the same result at this point, the NetScaler

switches to the round robin method, and continues to distribute connections using that method.

The following table explains how connections are distributed in the three-service load balancing setup described above.

Request Received	Service Selected	Current N Value (Number of Active Transactions * TTFB)	Remarks
Request-1	Service-HTTP-3 (N = 0)	N = 2	Service-HTTP-3 has the lowest N value.
Request-2	Service-HTTP-3 (N = 2)	N = 4	
Request-3	Service-HTTP-3 (N = 3)	N = 6	
Request-4	Service-HTTP-1 (N = 6)	N = 8	Service-HTTP-1 and Service-HTTP-3 have the same N values.
Request-5	Service-HTTP-3 (N = 6)	N = 8	
Request-6	Service-HTTP-2 (N = 7)	N = 8	Service-HTTP-2 has the lowest N value.
Request-7	Service-HTTP-1 (N = 8)	N = 15	Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 have the same N values.
Request-8	Service-HTTP-2 (N = 8)	N = 9	

Request Received	Service Selected	Current N Value (Number of Active Transactions) * TFB	Remarks

The virtual server selects a service based on the value (Nw) in the following expression:

Nw = (N) * (10000 / weight)

Suppose Service-HTTP-1 is assigned a weight of 2, Service-HTTP-2 is assigned weight of 3, and Service-HTTP-3 is assigned weight of 4.

The NetScaler appliance distributes requests as follows:

- Service-HTTP-3 receives the first request, because it is not handling any active transactions.

  If services are not handling any active transactions, the NetScaler selects them regardless of the weights assigned to them.

- Service-HTTP-3 receives the second, third, fourth, and fifth requests, because this service has the lowest Nw value.
- Service-HTTP-2 receives the sixth request, because this service has the lowest Nw value.
- Service-HTTP-3 receives the seventh request, because this service has the lowest Nw value.
- Service-HTTP-2 receives the eighth request, because this service has the lowest Nw value.

Service-HTTP-1 has the lowest weight and therefore the highest Nw value, so the virtual server does not select it for load balancing.

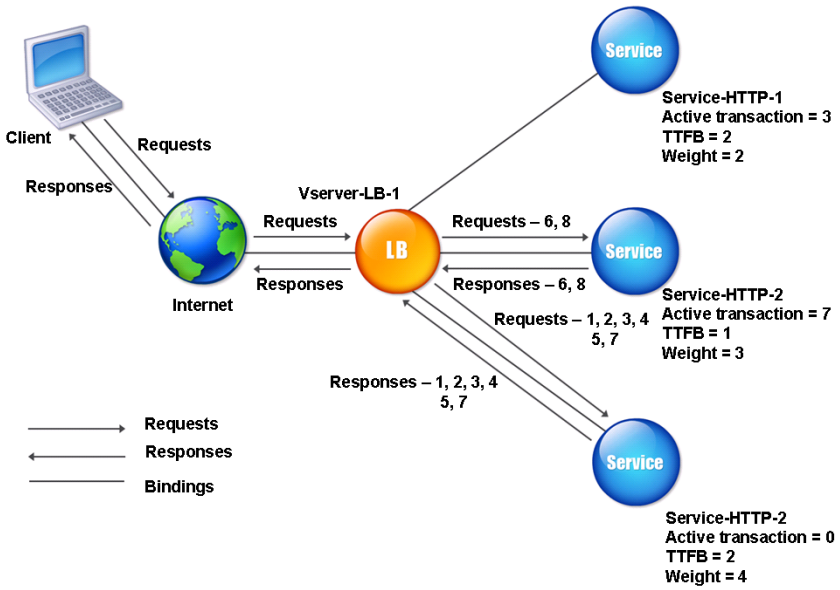
The following table explains how connections are distributed in the three-service load balancing setup described above.

Request Received	Service Selected	Current Nw Value (Number of Active Transactions) * (10000 / Weight)	Remarks
Request-1	Service-HTTP-3 (Nw = 0)	Nw = 2500	Service-HTTP-3 has the lowest Nw value.
Request-2	Service-HTTP-3 (Nw = 2500	Nw = 5000	
Request-3	Service-HTTP-3 (Nw = 5000)	Nw = 15000	
Request-4	Service-HTTP-3 (Nw = 15000)	Nw = 20000	

Request Received	Service Selected	Current Nw Value (Number of Active Transactions) * (10000 / Weight)	Remarks
Request-5	Service-HTTP-3 (Nw = 20000)	Nw = 25000	
Request-6	Service-HTTP-2 (Nw = 23333.34)	Nw = 26666.67	Service-HTTP-2 has the lowest Nw value.
Request-7	Service-HTTP-3 (Nw = 25000)	Nw= 30000	Service-HTTP-3 has the lowest Nw value.
Request-8	Service-HTTP-2 (Nw = 26666.67)	Nw = 33333.34	Service-HTTP-2 has the lowest Nw value.
Service-HTTP-1 is selected for load balancing when it completes its active transactions or when the Nw values of other services (Service-HTTP-2 and Service-HTTP-3) are equal to 105000.			

The following diagram illustrates how the NetScaler appliance uses the least response time method when weights are assigned.

Figure 2. How the Least Response Time Load Balancing Method Works When Weights Are Assigned



To configure the least response time method, see Configuring a Load Balancing Method that Does Not Include a Policy.

When a load balancing virtual server is configured to use the least response time method with monitors, it uses the existing monitoring infrastructure to select the service with the smallest number of active transactions and the fastest average response time. Before you use the least response time method with monitoring, you must bind application-specific monitors to each service and enable least response time method mode on these monitors. The NetScaler appliance then makes load balancing decisions based on the response times it calculates from monitoring probes. For more information about configuring monitors, see Configuring Monitors in a Load Balancing Setup.

You can use the least response time method with monitors to select non-HTTP and non-HTTPS services. You can also use this method when several monitors are bound to a service. Each monitor determines the response time by using the protocol that it measures for the service that it is bound to. The virtual server then calculates an average response time for that service by averaging the results.

The following table summarizes how response times are calculated for various monitors.

Monitor	Response Time Calculation
PING	Time difference between the ICMP ECHO request and the ICMP ECHO response.
TCP	Time difference between the SYN request and the SYN+ACK response.
HTTP	Time difference between the HTTP request (after the TCP connection is established) and the HTTP response.
TCP-ECV	Time difference between the time the data send string is sent and the data receive string is returned. A tcp-ecv monitor without the send and receive strings is considered to have an incorrect

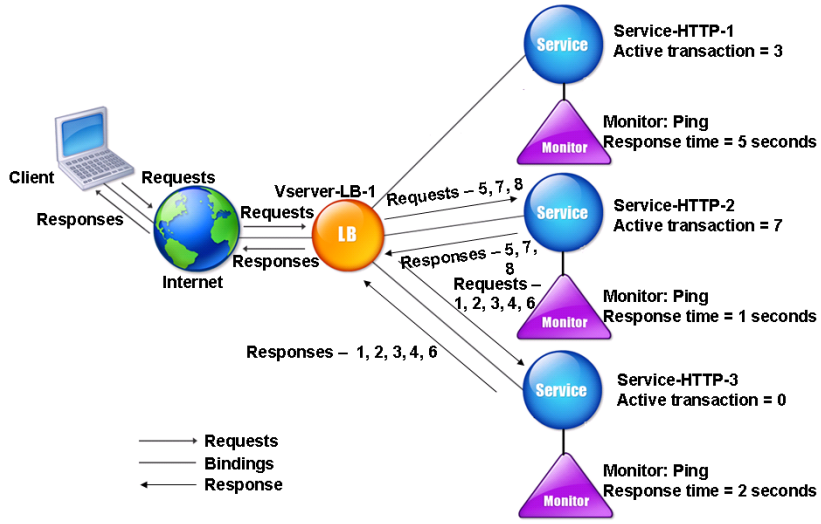
Monitor	Response Time Calculation
HTTP-ECV	Time difference between the HTTP request and the HTTP response.
UDP-ECV	Time difference between the UDP send string and the UDP receive string. A udp-ecv monitor without the receive string is considered to have an incorrect configuration.
DNS	Time difference between a DNS query and the DNS response.
TCPS	Time difference between a SYN request and the SSL handshake completion.
FTP	Time difference between the sending of the user name and the completion of user authentication.
HTTPS (monitors HTTPS requests)	Time difference is same as for the HTTP monitor.
HTTPS-ECV (monitors HTTPS requests)	Time difference is same as for the HTTP-ECV monitor
USER	Time difference between the time when a request is sent to the dispatcher and the time when the dispatcher response is received.

The following example shows how the NetScaler appliance selects a service for load balancing by using the least response time method with monitors. Consider the following three services:

- Service-HTTP-1 is handling 3 active transactions and the response time is five seconds.
- Service-HTTP-2 is handling 7 active transactions and the response time is one second.
- Service-HTTP-3 is not handling any active transactions and the response time is two seconds.

The following diagram illustrates the process that the NetScaler appliance follows when it forwards requests.

Figure 3. How the Least Response Time Load Balancing Method Works When Using Monitors



The virtual server selects a service by using the value (N) in the following expression:

N = Number of active transactions * Response time that is determined by the monitor

The virtual server delivers requests as follows:

- Service-HTTP-3 receives the first request, because this service is not handling any active transaction.
- Service-HTTP-3 receives the second, third, and fourth requests, because this service has the lowest N value.
- Service-HTTP-2 receives the fifth request, because this service has the lowest N value.
- Since both Service-HTTP-2 and Service-HTTP-3 currently have the same N value, the NetScaler appliance switches to the round robin method Therefore, Service-HTTP-3 receives the sixth request.
- Service-HTTP-2 receives the seventh and eighth requests, because this service has the lowest N value.

Service-HTTP-1 is not considered for load balancing, because it is more heavily loaded (has the highest N value) when compared to the other two services. However, if Service-HTTP-1 completes its active transactions, the NetScaler appliance again considers that service for load balancing.

The following table summarizes how N is calculated for the services.

Request Received	Service Selected	Current N Value (Number of Active Transactions)	Remarks
Request-1	Service-HTTP-3 (N = 0)	N = 2	Service-HTTP-3 has the lowest N value.
Request-2	Service-HTTP-3 (N = 2)	N = 4	

Request Received	Service Selected	Current N Value (Number of Active Transactions)	Remarks
Request-3	Service-HTTP-3 (N = 4)	N = 6	
Request-4	Service-HTTP-3 (N = 6)	N = 8	
Request-5	Service-HTTP-2 (N = 7)	N = 8	Service-HTTP-1 and Service-HTTP-3 have the same N values.
Request-6	Service-HTTP-3 (N = 8)	N = 10	
Request-7	Service-HTTP-2 (N = 8)	N = 9	Service-HTTP-2 has the lowest N value.
Request-8	Service-HTTP-1 (N = 9)	N = 10	
Service-HTTP-1 is again selected for load balancing when it completes its active transactions or when the N value of the other services (Service-HTTP-2 and Service-HTTP-3) is equal to 15.			

The NetScaler appliance also performs load balancing by using the number of active transactions, response time, and weights if different weights are assigned to services. The NetScaler appliance selects the service by using the value (Nw) in the following expression:

Nw = (N) * (10000 / weight)

As in the preceding example, suppose Service-HTTP-1 is assigned a weight of 2, Service-HTTP-2 is assigned a weight of 3, and Service-HTTP-3 is assigned a weight of 4.

The NetScaler appliance delivers requests as follows:

- Service-HTTP-3 receives the first request, because it is not handling any active transactions.
- Service-HTTP-3 receives the second, third, and fourth requests, because this service has the lowest Nw value.

- Service-HTTP-2 receives the fifth request, because this service has the lowest Nw value.
- Service-HTTP-3 receives the sixth request, because this service has the lowest Nw value.
- Service-HTTP-2 receives the seventh and the eighth requests, because this service has the lowest Nw value.

Service-HTTP-1 has the lowest weight and the highest Nw value, so the NetScaler appliance does not select it for load balancing.

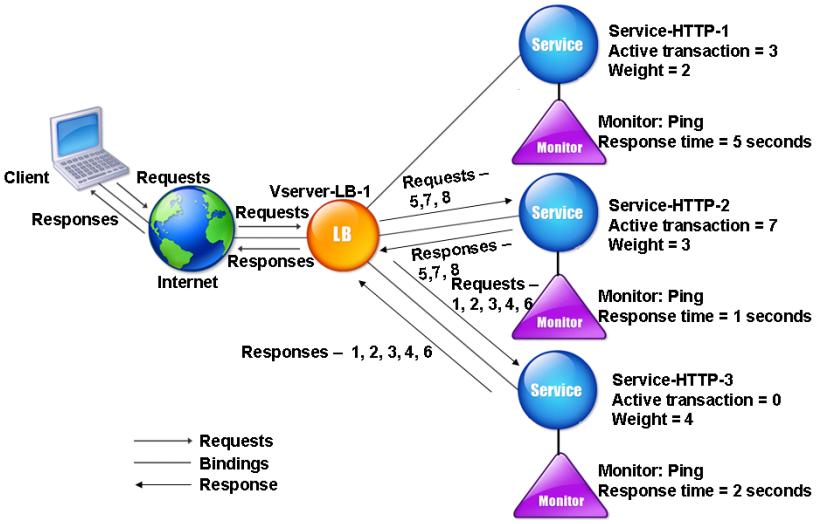
The following table summarizes how Nw is calculated for various monitors.

Request Received	**Service Selected**	**Current Nw Value (Number of Active Transactions) * (10000 / Weight)**	**Remarks**
Request-1	Service-HTTP-3 (Nw = 0)	Nw = 5000	Service-HTTP-3 has the lowest Nw value.
Request-2	Service-HTTP-3 (Nw = 5000	Nw = 10000	
Request-3	Service-HTTP-3 (Nw = 15000)	Nw = 20000	
Request-4	Service-HTTP-3 (Nw = 20000)	Nw = 25000	
Request-5	Service-HTTP-2 (Nw = 23333.34)	Nw = 26666.67	Service-HTTP-2 has the lowest Nw value.
Request-6	Service-HTTP-3 (Nw = 25000)	Nw = 30000	Service-HTTP-3 has the lowest Nw value.

Request Received	Service Selected	Current Nw Value (Number of Active Transactions) * (10000 / Weight)	Remarks
Request-7	Service-HTTP-2 (Nw = 23333.34)	Nw= 26666.67	Service-HTTP-2 has the lowest Nw value.
Request-8	Service-HTTP-2 (Nw =25000)	Nw = 30000	Service-HTTP-2 has the lowest Nw value.
Service-HTTP-1 is selected for load balancing when it completes its active transactions or when the Nw value of the other services (Service-HTTP-2 and Service-HTTP-3) is equal to 75000.			

The following diagram illustrates how the virtual server uses the least response time method when weights are assigned.

Figure 4. How the Least Response Time Load Balancing Method with Monitors Works When Weights Are Assigned



To configure the least response time method using monitors, see Configuring a Load Balancing Method that Does Not Include a Policy.

About Hashing Methods

Mar 16, 2012

Load balancing methods based on hashes of certain connection information or header information constitute the majority of the NetScaler appliance's load balancing methods. Hashes are shorter and easier to use than the information that they are based on, while retaining enough information to ensure that no two different pieces of information generate the same hash and are therefore confused with one another.

You can use the hashing load balancing methods in an environment where a cache serves a wide range of content from the Internet or specified origin servers. Caching requests reduces request and response latency, and ensures better resource (CPU) utilization, making caching popular on heavily used Web sites and application servers. Since these sites also benefit from load balancing, hashing load balancing methods are widely useful.

The NetScaler provides the following hashing methods:

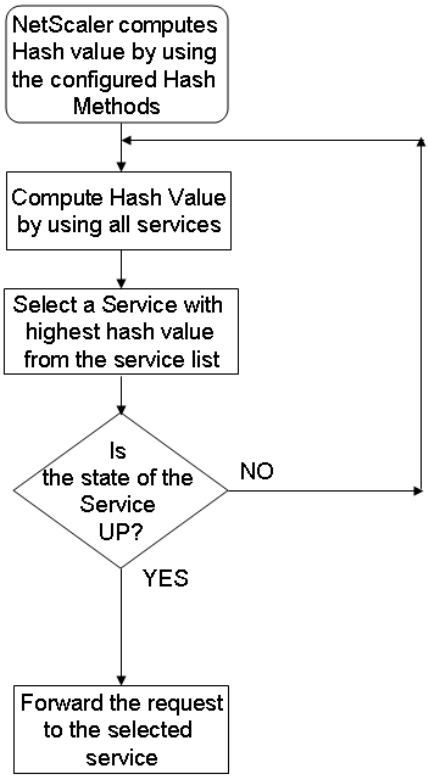
- URL hash method
- Domain hash method
- Destination IP hash method
- Source IP hash method
- Source IP Destination IP hash method
- Source IP Source Port hash method
- Call ID hash method
- Token method

These hashing algorithms ensure minimal disruption when services are added to or deleted from your load balancing setup. Most of them calculate two hash values:

- A hash of the service's IP address and port.
- A hash of the incoming URL, the domain name, the source IP address, the destination IP address, or the source and destination IP addresses, depending on the configured hash method.

The NetScaler appliance then generates a new hash value by using both of those hash values. Finally, it forwards the request to the service with highest hash value. As the appliance computes a hash value for each request and selects the service that will process the request, it populates a cache. Subsequent requests with the same hash value are sent to the same service. The following flow chart illustrates this process.

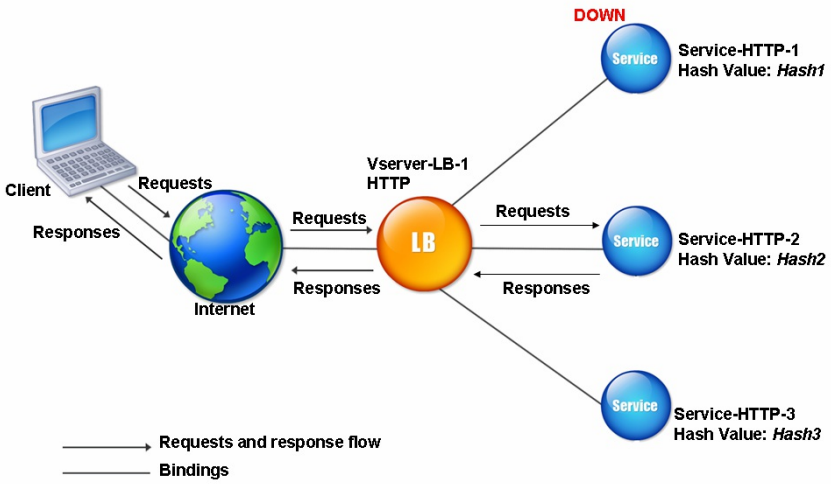
Figure 1. How the Hashing Methods Distribute Requests



Hashing methods can be applied to IPv4 and IPv6 addresses.

Consider a scenario where three services (Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3) are bound to a virtual server, any hash method is configured, and the hash value is Hash1. When the configured services are UP, the request is sent to Service-HTTP-1. If Service-HTTP-1 is down, the NetScaler appliance calculates the hash value for the last log of the number of services. The NetScaler then selects the service with the highest hash value, such as Service-HTTP-2. The following diagram illustrates this process.

Figure 2. Entity Model for Hashing Methods



Note: If the NetScaler appliance fails to select a service by using a hashing method, it defaults to the least connection method to select a service for the incoming request. You should adjust server pools by removing services during periods of low traffic to enable the caches to repopulate without affecting performance on your load balancing setup.

The URL Hash Method

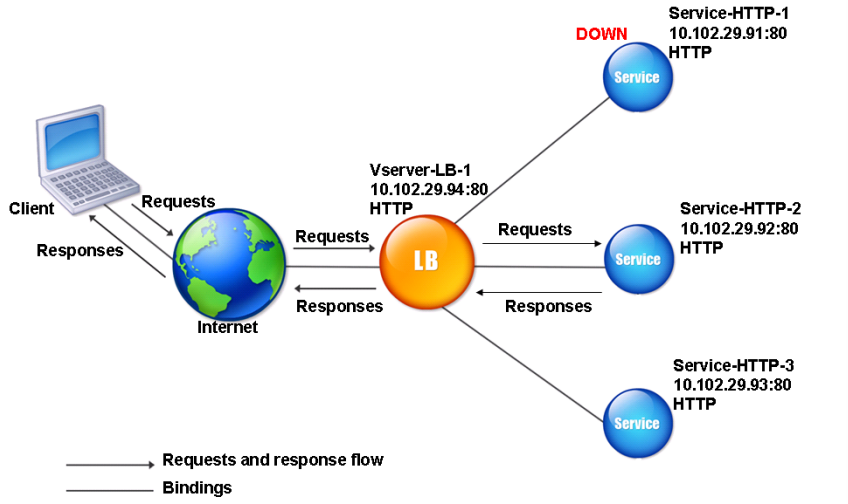
When you configure the NetScaler appliance to use the URL hash method for load balancing the services, for selecting a service, the NetScaler generates a hash value of the HTTP URL present in the incoming request. If the service selected by the hash value is DOWN, the algorithm has a method to select another service from the list of active services. The NetScaler caches the hashed value of the URL, and when it receives subsequent requests that use the same URL, it forwards them to the same service. If the NetScaler cannot parse an incoming request, it uses the round robin method for load balancing instead of the URL hash method.

For generating the hash value, NetScaler uses a specific algorithm and considers a part of the URL. By default, the NetScaler considers the first 80 bytes of the URL. If the URL is of less than 80 bytes, the complete URL is used. You can specify a different length. The hash length can be from 1 to 4096 bytes. Generally, if long URLs are used where only a small number of characters are different, it is a good idea to make the hash length as high as possible to try to ensure a more even load distribution.

Consider a scenario where three services, Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3, are bound to a virtual server, and the load balancing method configured on the virtual server is the URL hash method. The virtual server receives a request and the hash value of the URL is U1. NetScaler selects Service-HTTP-1. If Service-HTTP-1 is DOWN, the NetScaler selects Service-HTTP-2.

The following diagram illustrates this process.

Figure 3. How URL Hashing Operates



If both Service-HTTP-1 and Service-HTTP-2 are DOWN, NetScaler sends requests with hash value U1 to Service-HTTP-3.

If Service-HTTP-1 and Service-HTTP-2 are down, requests that generate the hash URL1 are sent to Service-HTTP-3. If these services are UP, requests that generate the hash URL1 are distributed in the following manner:

- If the Service-HTTP-2 is up, the request is sent to Service-HTTP-2.
- If the Service-HTTP-1 is up, the request is sent to Service-HTTP-1.
- If Service-HTTP-1 and Service-HTTP-2 are up at the same time, the request is sent to Service-HTTP-1.

To configure the URL hash method, see Configuring a Load Balancing Method that Does Not Include a Policy. Select the load balancing method as URL Hash, and set the hash length to the number of bytes to be used for generating the hash

value.

The Domain Hash Method

A load balancing virtual server configured to use the domain hash method uses the hashed value of the domain name in the HTTP request to select a service. The domain name is taken from either the incoming URL or the Host header of the HTTP request. If the domain name appears in both the URL and the Host header, the NetScaler gives preference to the URL.

If you configure domain name hashing, and an incoming HTTP request does not contain a domain name, the NetScaler appliance defaults to the round robin method for that request.

The hash-value calculation uses the name length or hash length value, whichever is smaller. By default, the NetScaler appliance calculates the hash value from the first 80 bytes of the domain name. To specify a different number of bytes in the domain name when calculating the hash value, you can set the hashLength parameter (Hash Length in the configuration utility) to a value of from 1 to 4096 (bytes).

To configure the domain hash method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Destination IP Hash Method

Updated: 2013-09-03

A load balancing virtual server configured to use the destination IP hash method uses the hashed value of the destination IP address to select a server. You can mask the destination IP address to specify which part of it to use in the hash value calculation, so that requests that are from different networks but destined for the same subnet are all directed to the same server. This method supports IPv4 and IPv6-based destination servers.

This load balancing method is appropriate for use with the cache redirection feature.

To configure the destination IP hash method for an IPv4 destination server, you set the netMask parameter. To configure this method for an IPv6 destination server, you use the v6NetMaskLen parameter. In the configuration utility, text boxes for setting these parameters appear when you select the Destination IP Hash method.

To configure the destination IP hash method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Source IP Hash Method

A load balancing virtual server configured to use the source IP hash method uses the hashed value of the client IPv4 or IPv6 address to select a service. To direct all requests from source IP addresses that belong to a particular network to a specific destination server, you must mask the source IP address. For IPv4 addresses, use the netMask parameter. For IPv6 addresses, use the v6NetMaskLength parameter.

To configure the source IP hash method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Source IP Destination IP Hash Method

A load balancing virtual server configured to use the source IP destination IP hash method uses the hashed value of the source and destination IP addresses (either IPv4 or IPv6) to select a service. Hashing is symmetric; the hash-value is the same regardless of the order of the source and destination IPs. This ensures that all packets flowing from a particular client to the same destination are directed to the same server.

To direct all requests that belong to a particular network to a specific destination server, you must mask the source IP address. For IPv4 addresses, use the netMask parameter. For IPv6 addresses, use the v6NetMaskLength parameter.

To configure the source IP destination IP hash method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Source IP Source Port Hash Method

A load balancing virtual server configured to use the source IP source port hash method uses the hash value of the source IP (either IPv4 or IPv6) and source port to select a service. This ensures that all packets on a particular connection are directed to the same service.

This method is used in connection mirroring and firewall load balancing. For more information about connection mirroring, see Connection Failover.

To direct all requests that belong to a particular network to a specific destination server, you must mask the source IP address. For IPv4 addresses, use the netMask parameter. For IPv6 addresses, use the v6NetMaskLength parameter.

To configure the source IP source port hash method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Call ID Hash Method

A load balancing virtual server configured to use the call ID hash method uses the hash value of the call ID in the SIP header to select a service. Packets for a particular SIP session are therefore always directed to the same proxy server.

This method is applicable to SIP load balancing. For more information about SIP load balancing, see Monitoring SIP Services.

To configure the call ID hash method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Least Bandwidth Method

Jan 17, 2014

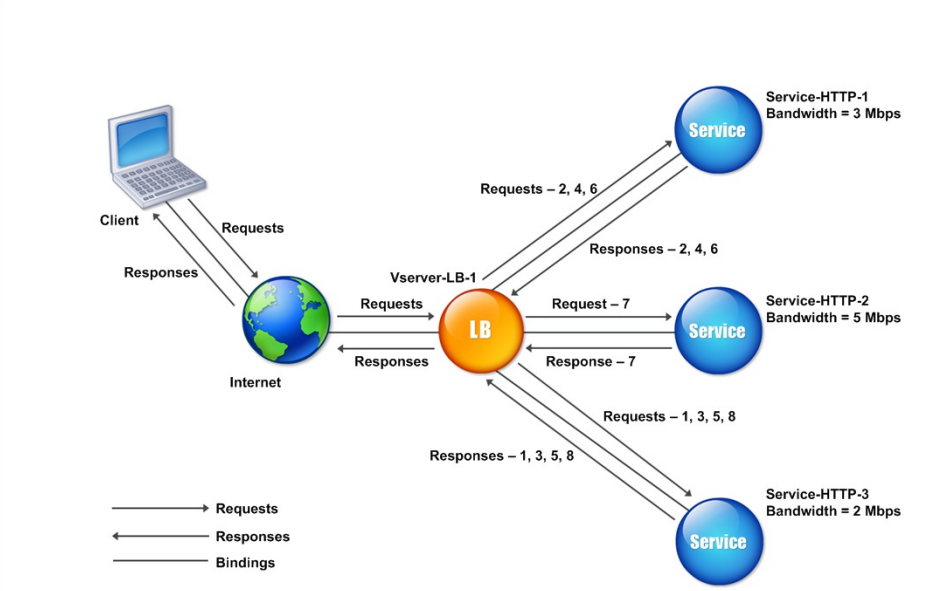
A load balancing virtual server configured to use the least bandwidth method selects the service that is currently serving the least amount of traffic, measured in megabits per second (Mbps). The following example shows how the virtual server selects a service for load balancing by using the least bandwidth method.

Consider three services, Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3.

- Service-HTTP-1 has 3 Mbps bandwidth.
- Service-HTTP-2 has 5 Mbps bandwidth.
- Service-HTTP-3 has 2 Mbps bandwidth.

The following diagram illustrates how the virtual server uses the least bandwidth method to forward requests to the three services.

Figure 1. How the Least Bandwidth Load Balancing Method Works



The virtual server selects the service by using the bandwidth value (N), which is the sum of the number of bytes transmitted and received over the previous 14 seconds. If each request requires 1 Mbps bandwidth, the NetScaler appliance delivers requests as follows:

- Service-HTTP-3 receives the first request, because this service has the lowest N value.
- Since Service-HTTP-1 and Service-HTTP-3 now have same N value, the virtual server switches to the round robin method for these servers, alternating between them. Service-HTTP-1 receives the second request, Service-HTTP-3 receives the third request, Service-HTTP-1 receives the fourth request, Service-HTTP-3 receives the fifth request, and Service-HTTP-1 receives the sixth request.
- Since Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 now all have same N value, the virtual server includes Service-HTTP-2 in the round robin list. Therefore, Service-HTTP-2 receives the seventh request, Service-HTTP-3 receives the eighth request, and so on.

The following table summarizes how N is calculated.

Request Received	Service Selected	Current N Value	Remarks
Request-1	Service-HTTP-3 (N = 2)	N = 3	Service-HTTP-3 has the lowest N value.
Request-2	Service-HTTP-1 (N = 3)	N = 4	Service-HTTP-1 and Service-HTTP-3 have the same N values.
Request-3	Service-HTTP-3 (N = 3)	N = 4	
Request-4	Service-HTTP-1 (N = 4)	N = 5	
Request-5	Service-HTTP-3 (N = 4)	N = 5	
Request-6	Service-HTTP-1 (N = 5)	N = 6	Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 have the same N values.
Request-7	Service-HTTP-2 (N = 5)	N = 6	
Request-8	Service-HTTP-3 (N = 5)	N = 6	

Note: If you enable the RTSP NAT option on the virtual server, the NetScaler appliance uses the number of data and control bytes exchanged to determine the bandwidth usage for RTSP services. For more information about RTSP NAT

option, see Managing RTSP Connections.
The NetScaler appliance also performs load balancing by using the bandwidth and weights if different weights are assigned to the services. It selects a service by using the value (Nw) in the following expression:

Nw = (N) * (10000 / weight)

As in the preceding example, suppose Service-HTTP-1 is assigned a weight of 2, Service-HTTP-2 is assigned a weight of 3, and Service-HTTP-3 is assigned a weight of 4. The NetScaler appliance delivers requests as follows:

- Service-HTTP-3 receives the first second, third, fourth, and fifth requests, because this service has the lowest Nw value.
- Service-HTTP-1 receives the sixth request, because this service has the lowest Nw value.
- Service-HTTP-3 receives the seventh request, because this service has the lowest Nw value.
- Service-HTTP-2 receives the eighth request, because this service has the lowest Nw value.

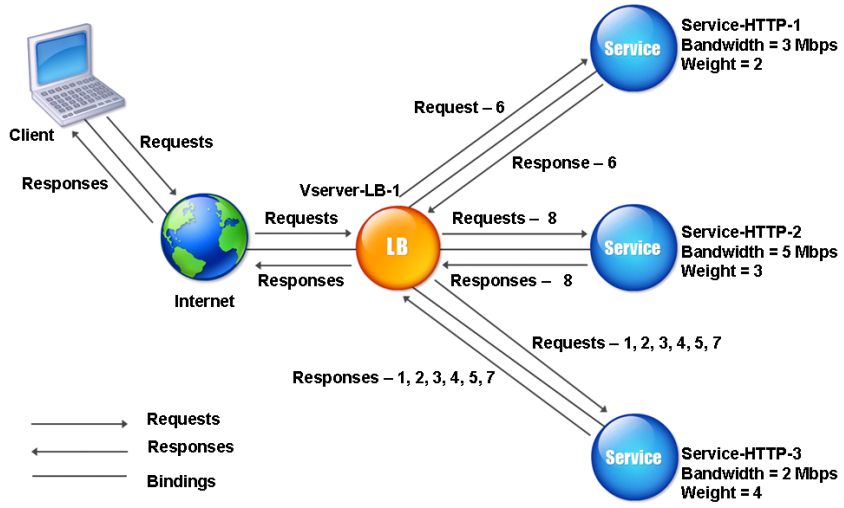
The following table summarizes how Nw is calculated.

Request Received	**Service Selected**	**Current Nw Value (Number of Active Transactions) * (10000 / Weight)**	**Remarks**
Request-1	Service-HTTP-3 (Nw = 5000)	Nw = 5000	Service-HTTP-3 has the lowest Nw value.
Request-2	Service-HTTP-3 (Nw = 5000)	Nw = 7500	
Request-3	Service-HTTP-3 (Nw = 7500)	Nw = 10000	
Request-4	Service-HTTP-3 (Nw = 10000)	Nw = 12500	
Request-5	Service-HTTP-3 (Nw =	Nw = 15000	

Request Received	Service Selected	Current Nw Value (Number of Active Transactions) * (10000 / Weight)	Remarks
Request-6	Service-HTTP-1 (Nw = 15000)	Nw = 20000	Service-HTTP-1 and Service-HTTP-3 have the same Nw value.
Request-7	Service-HTTP-3 (Nw = 15000)	Nw = 17500	
Request-8	Service-HTTP-2 (Nw = 16666.67)	Nw = 20000	Service-HTTP-2 has the lowest Nw value.

The following diagram illustrates how the virtual server uses the least bandwidth method when weights are assigned to the services.

Figure 2. How the Least Bandwidth Load Balancing Method Works When Weights Are Assigned



To configure the least bandwidth method, see Configuring a Load Balancing Method that Does Not Include a Policy.

The Least Packets Method

Jan 17, 2014

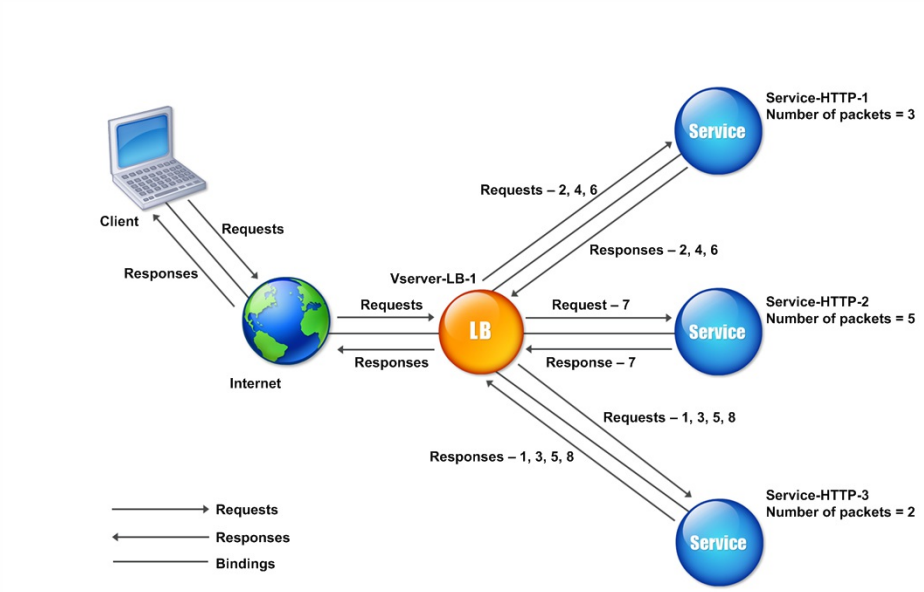
A load balancing virtual server configured to use the least packets method selects the service that has received the fewest packets in the last 14 seconds.

For example, consider three services, Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3.

- Service-HTTP-1 has handled three packets in last 14 seconds.
- Service-HTTP-2 has handled five packets in last 14 seconds.
- Service-HTTP-3 has handled two packets in last 14 seconds.

The following diagram illustrates how the NetScaler appliance uses the least packets method to choose a service for each request that it receives.

Figure 1. How the Least Packets Load Balancing Method Works



The NetScaler appliance selects a service by using the number of packets (N) transmitted and received by each service in the last 14 seconds. Using this method, it delivers requests as follows:

- Service-HTTP-3 receives the first request, because this service has the lowest N value.
- Since Service-HTTP-1 and Service-HTTP-3 now have the same N value, the virtual server switches to the round robin method. Service-HTTP-1 therefore receives the second request, Service-HTTP-3 receives the third request, Service-HTTP-1 receives the fourth request, Service-HTTP-3 receives the fifth request, and Service-HTTP-1 receives the sixth request.
- Since Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 all now have same N value, the virtual server switches to the round robin method for Service-HTTP-2 as well, including it in the round robin list. Therefore, Service-HTTP-2 receives the seventh request, Service-HTTP-3 receives the eighth request, and so on.

The following table summarizes how N is calculated.

Request Received	Service Selected	Current N Value	Remarks
Request-1	Service-HTTP-3 (N = 2)	N = 3	Service-HTTP-3 has the lowest N value.
Request-2	Service-HTTP-1 (N = 3)	N = 4	Service-HTTP-1 and Service-HTTP-3 have the same N values.
Request-3	Service-HTTP-3 (N = 3)	N = 4	
Request-4	Service-HTTP-1 (N = 4)	N = 5	
Request-5	Service-HTTP-3 (N = 4)	N = 5	
Request-6	Service-HTTP-1 (N = 5)	N = 6	Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 have the same N values.
Request-7	Service-HTTP-2 (N = 5)	N = 6	
Request-8	Service-HTTP-3 (N = 5)	N = 6	

Note: If you enable the RTSP NAT option on the virtual server, the NetScaler uses the number of data and control packets to calculate the number of packets for RTSP services. For more information about RTSP NAT option, see Managing RTSP

Connections.

The NetScaler appliance also performs load balancing by using the number of packets and weights when a different weight is assigned to each service. It selects a service by using the value (Nw) in the following expression:

Nw = (N) * (10000 / weight)

As in the preceding example, suppose Service-HTTP-1 is assigned a weight of 2, Service-HTTP-2 is assigned a weight of 3, and Service-HTTP-3 is assigned a weight of 4. The NetScaler appliance delivers requests as follows:

- Service-HTTP-3 receives the first second, third, fourth, and fifth requests, because this service has the lowest Nw value.
- Service-HTTP-1 receives the sixth request, because this service has the lowest Nw value.
- Service-HTTP-3 receives the seventh request, because this service has the lowest Nw value.
- Service-HTTP-2 receives the eighth request, because this service has the lowest Nw value.

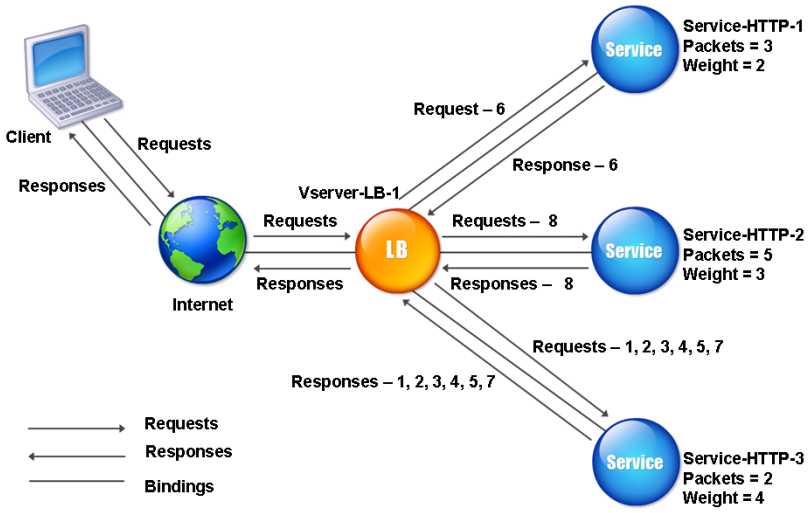
The following table summarizes how Nw is calculated.

Request Received	**Service Selected**	**Current Nw Value (Number of Active Transactions) * (10000 / weight)**	**Remarks**
Request-1	Service-HTTP-3 (Nw = 5000)	Nw = 5000	Service-HTTP-3 has the lowest Nw value.
Request-2	Service-HTTP-3 (Nw = 5000)	Nw = 7500	
Request-3	Service-HTTP-3 (Nw = 7500)	Nw = 10000	
Request-4	Service-HTTP-3 (Nw = 10000)	Nw = 12500	
Request-5	Service-HTTP-3 (Nw =	Nw = 15000	

Request Received	Service Selected	Current Nw Value (Number of Active Transactions) * (10000 / weight)	Remarks
Request-6	Service-HTTP-1 (Nw = 15000)	Nw = 20000	Service-HTTP-1 and Service-HTTP-3 have the same Nw value.
Request-7	Service-HTTP-3 (Nw = 15000)	Nw = 17500	
Request-8	Service-HTTP-2 (Nw = 16666.67)	Nw = 20000	Service-HTTP-2 has the lowest Nw value.

The following diagram illustrates how the virtual server uses the least packets method when weights are assigned.

Figure 2. How the Least Packets Method Works When Weights Are Assigned



To configure the least packets method, see Configuring a Load Balancing Method that Does Not Include a Policy.

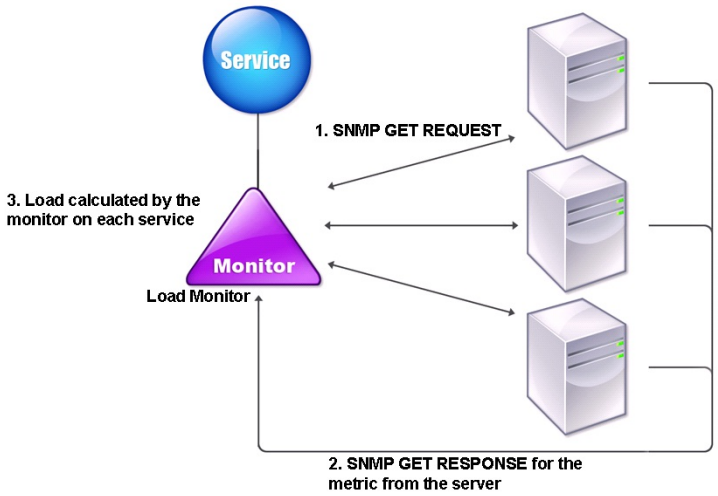
The Custom Load Method

Feb 24, 2015

Custom load balancing is performed on server parameters such as CPU usage, memory, and response time. When using the custom load method, the NetScaler appliance usually selects a service that is not handling any active transactions. If all of the services in the load balancing setup are handling active transactions, the appliance selects the service with the smallest load. A special type of monitor, known as a load monitor, calculates the load on each service in the network. The load monitors do not mark the state of a service, but they do take services out of the load balancing decision when those services are not UP.

For more information about load monitors, see Understanding Load Monitors. The following diagram illustrates how a load monitor operates.

Figure 1. How Load Monitors Operate



The load monitor uses Simple Network Management Protocol (SNMP) probes to calculate load on each service by sending an SNMP GET request to the service. This request contains one or more object IDs (OIDs). The service responds with an SNMP GET response, with metrics corresponding to the SNMP OIDs. The load monitor uses the response metrics, described below, to calculate the load on the service.

The load monitor calculates the load on a service by using the following parameters:

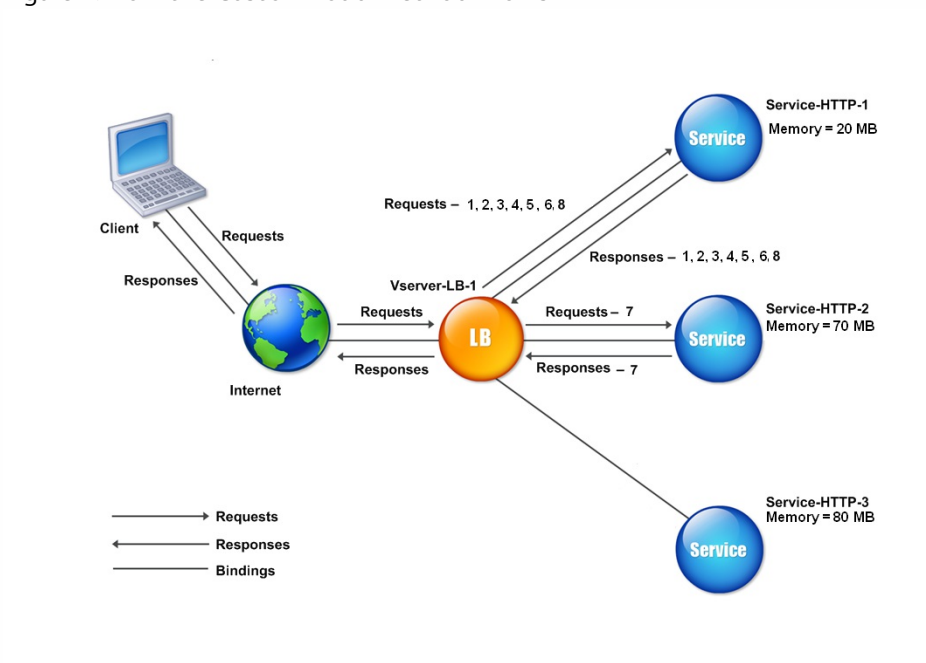
- Metrics values retrieved through SNMP probes that exist as tables in the NetScaler.
- Threshold value set for each metric.
- Weight assigned to each metric.

For example, consider three services, Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3.

- Service-HTTP-1 is using 20 megabytes (MB) of memory.
- Service-HTTP-2 is using 70 MB of memory.
- Service-HTTP-3 is using 80 MB of memory.

The load balanced servers can export metrics such as CPU and memory usage to the services, which can in turn provide them to the load monitor. The load monitor sends an SNMP GET request containing the OIDs 1.3.6.1.4.1.5951.4.1.1.41.1.5, 1.3.6.1.4.1.5951.4.1.1.41.1.4, and 1.3.6.1.4.1.5951.4.1.1.41.1.3 to the services. SNMP OIDs of type STRING are not supported, because you cannot calculate the load by using a STRING OID. Loads can be calculated by using other data types, such as INT and gauge32. The three services respond to the request. The NetScaler appliance compares the exported metrics, and then selects Service-HTTP-1 because it has more available memory. The following diagram illustrates this process.

Figure 2. How the Custom Load Method Works



If each request uses 10 MB memory, the NetScaler appliance delivers requests as follows:

- Service-HTTP-1 receives the first, second, third, fourth, and fifth requests, because this service has the lowest N value.
- Service-HTTP-1 and Service-HTTP-2 now have the same load, so the virtual server reverts to the round robin method for these servers. Therefore, Service-HTTP-2 receives the sixth request, and Service-HTTP-1 receives the seventh request.
- Since Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 all now have same load, the virtual server reverts to the round robin method for Service-HTTP-3 as well. Therefore, Service-HTTP-3 receives the eighth request.

The following table summarizes how N is calculated.

Request received	Service selected	Current N Value (Number of Active Transactions)	Remarks
Request-1	Service-HTTP-1 (N = 20)	N = 30	Service-HTTP-3 has the lowest N value.
Request-2	Service-HTTP-1 (N = 30)	N = 40	

Request received	Service selected	Current N Value (Number of Active Transactions)	Remarks
Request-3	Service-HTTP-1 (N = 40)	N = 50	
Request-4	Service-HTTP-1 (N = 50)	N = 60	
Request-5	Service-HTTP-1 (N = 60)	N = 70	
Request-6	Service-HTTP-1 (N = 70)	N = 80	Service-HTTP-2 and Service-HTTP-3 have the same N values.
Request-7	Service-HTTP-2 (N = 70)	N = 80	
Request-8	Service-HTTP-1 (N = 80)	N = 90	Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 have the same N values.

If different weights are assigned to the services, the custom load algorithm considers both the load on each service and the weight assigned to each service. It selects a service by using the value (Nw) in the following expression:

Nw = (N) * (10000 / weight)

As in the preceding example, suppose Service-HTTP-1 is assigned a weight of 4, Service-HTTP-2 is assigned a weight of 3, and Service-HTTP-3 is assigned a weight of 2. If each request uses 10 MB memory, the NetScaler appliance delivers requests as follows:

- Service-HTTP-1 receives the first, second, third, fourth, fifth, sixth, seventh, and eighth requests, because this service has the lowest Nw value.
- Service-HTTP-2 receives the ninth request, because this service has the lowest Nw value.

Service-HTTP-3 has the highest Nw value, and is therefore not considered for load balancing.

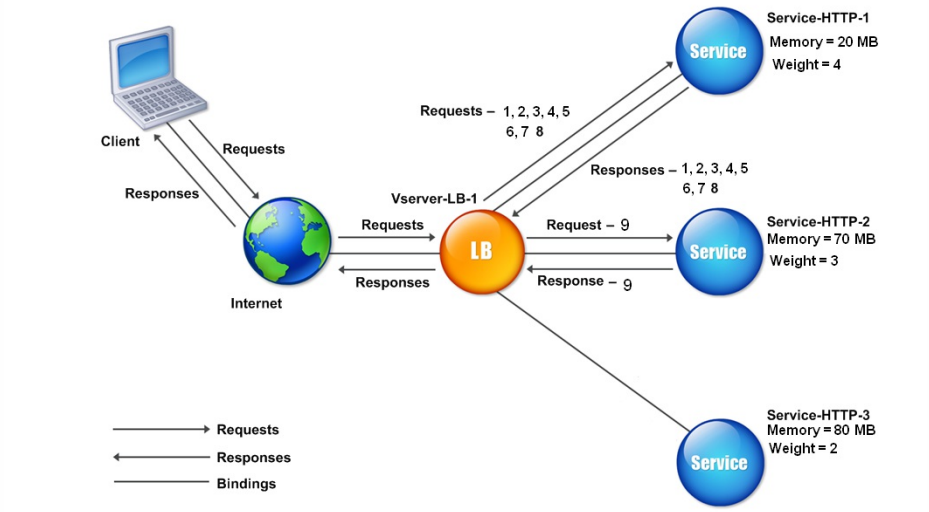
The following table summarizes how Nw is calculated.

Request received	Service selected	Current Nw Value (Number of Active Transactions) * (10000 / Weight)	Remarks
Request-1	Service-HTTP-1 (Nw = 50000)	Nw = 75000	Service-HTTP-1 has the lowest Nw value.
Request-2	Service-HTTP-1 (Nw = 5000)	Nw = 100000	
Request-3	Service-HTTP-1 (Nw = 15000)	Nw = 125000	
Request-4	Service-HTTP-1 (Nw = 20000)	Nw = 150000	
Request-5	Service-HTTP-1 (Nw = 23333.34))	Nw = 175000	
Request-6	Service-HTTP-1 (Nw = 25000)	Nw = 200000	
Request-7	Service-HTTP-1 (Nw = 23333.34)	Nw = 225000	
Request-8	Service-	Nw = 250000	

Request received	Service selected	Current Nw Value (Number of Active Transactions) * (10000 / Weight)	Remarks
	HTTP-1 (Nw = 25000)		
Request-9	Service-HTTP-2 (Nw = 233333.34)	Nw = 266666.67	Service-HTTP-2 has the lowest Nw value.
Service-HTTP-1 is selected for load balancing when it completes its active transactions or when the Nw value of other services (Service-HTTP-2 and Service-HTTP-3) is equal to 400,000.			

The following diagram illustrates how the NetScaler appliance uses the custom load method when weights are assigned.

Figure 3. How the Custom Load Method Works When Weights Are Assigned



To configure the custom load method, see Configuring a Load Balancing Method that Does Not Include a Policy.

Configuring the Token Method

Jul 15, 2014

A load balancing virtual server configured to use the token method bases its selection of a service on the value of a data segment extracted from the client request. The data segment is called the token. You configure the location and size of the token. For subsequent requests with the same token, the virtual server chooses the same service that handled the initial request.

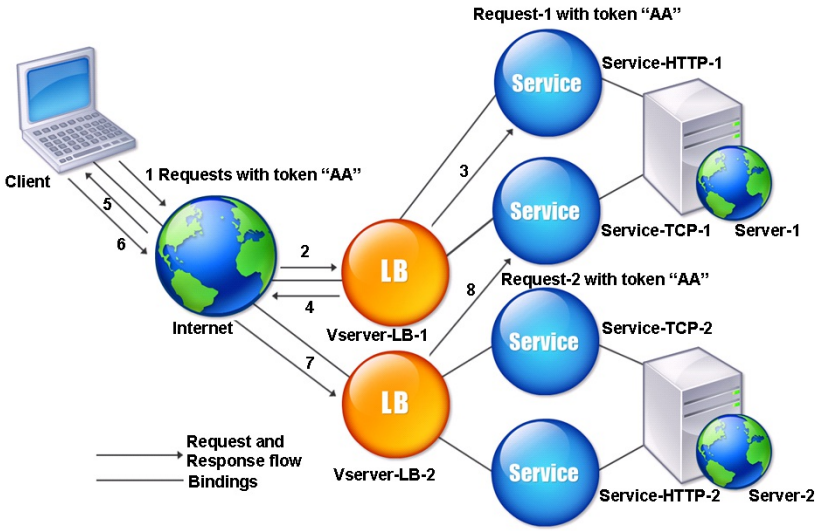
This method is content aware; it operates differently for TCP, HTTP, and HTTPS connections. For HTTP or HTTPS services, the token is found in the HTTP headers, the URL, or the BODY. To locate the token, you specify or create a classic or advanced expression. For more information on classic or advanced expressions, see Policy Configuration and Reference.

For HTTP services, the virtual server searches for the configured token in the first 24 kilobytes (KB) of the TCP payload. For non-HTTP (TCP, SSL, and SSL_TCP) services, the virtual server searches for the configured token in the first 16 packets if the total size of the 16 packets is less than 24 KB. But if the total size of the 16 packets is greater than 24 KB, the NetScaler searches for the token in the first 24 KB of payload. You can use this load balancing method across virtual servers of different types to make sure that requests presenting the same token are directed to appropriate services, regardless of the protocol used.

For example, consider a load balancing setup consisting of servers that contain Web content. You want to configure the NetScaler appliance to search for a specific string (the token) inside the URL query portion of the request. Server-1 has two services, Service-HTTP-1 and Service-TCP-1, and Server-2 has two services, Service-HTTP-2 and Service-TCP-2. The TCP services are bound to Vserver-LB-2, and the HTTP services are bound to Vserver-LB-1.

If Vserver-LB-1 receives a request with the token AA, it selects the service Service-HTTP-1 (bound to server-1) to process the request. If Vserver-LB-2 receives a different request with the same token (AA), it directs this request to the service Service-TCP-1. The following diagram illustrates this process.

Figure 1. How the Token Method Works



To configure the Token load balancing method by using the command line interface

At the command prompt, type the following commands to configure the token load balancing method and verify the configuration:

- set lb vserver <name> -lbMethod TOKEN -rule <rule> -datalength <length> -dataoffset <offset>
- show lb vserver <name>

Example

```
set lb vserver LB-VServer-1 -lbMethod TOKEN -rule 'AA' -datalength 2 -dataoffset 25

show lb vserver LB-VServer-1
```

To configure the Token load balancing method by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure a rule, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Method and Persistence tab and under LB Method, select Token.
4. Click Configure next to the Rule text box.
5. In the Create Expression dialog box, select Classic Syntax or Advanced Syntax.
6. Under Expression, click Add.
7. In the Add Expression dialog box, enter an expression. For more information about expressions, see Policy Configuration and Reference. For example, if you are configuring a classic expression, you can select an Expression Type of General, a Flow Type of REQ, a Protocol of HTTP, a Qualifier of URLQUERY, an Operator of CONTAINS, and in the Value text box, type AA.
8. Click OK, and then click Close.
9. In the Create Expression dialog box, click Create. The expression you created appears in the Rule text box.
10. Click OK.

Configuring a Load Balancing Method That Does Not Include a Policy

Jul 16, 2014

After you select a load balancing algorithm for your load balancing setup, you must configure the NetScaler appliance to use that algorithm. You can configure it by using the NetScaler command line or by using the configuration utility.

Note:
The token method is policy based and requires more configuration than is described here. To configure the token method, see Configuring the Token Method.

For some hash-based methods, you can mask an IP address to direct requests belonging to the same subnet to the same server. For more information, see The Destination IP Hash Method, The Source IP Hash Method, The Source IP Destination IP Hash Method, and The Source IP Source Port Hash Method.

To set the load balancing method by using the command line interface

At the command prompt, type:

set lb vserver <name> -lbMethod <method>

Example

```
set lb vserver Vserver-LB-1 -lbMethod LeastConnection
```

To set the load balancing method by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure an LB method, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Method and Persistence tab.
4. From the drop-down menu under LB Method, select a method, (for example, Least Response Time).
5. Click OK.

Persistence and Persistent Connections

Mar 16, 2012

Unless you configure persistence, a load balancing stateless protocol, such as HTTP, disrupts the maintenance of state information about client connections. Different transmissions from the same client might be directed to different servers even though all of the transmissions are part of the same session. You must configure persistence on a load balancing virtual server that handles certain types of Web applications, such as shopping cart applications.

Before you can configure persistence, you need to understand the different types of persistence, how they are used, and what the implications of each type is. You then need to configure the NetScaler appliance to provide persistent connections for those Web sites and Web applications that require them.

You can also configure backup persistence, which takes effect in the event that the primary type of persistence configured for a load balancing virtual server fails. You can configure persistence groups, so that a client transmission to any virtual server in a group can be directed to a server that has received previous transmissions from the same client.

For information about persistence with RADIUS load balancing, see Configuring RADIUS Load Balancing with Persistence.

About Persistence

Mar 16, 2012

You can choose from among any of several types of persistence for a given load balancing virtual server, which then routes to the same service all connections from the same user to your shopping cart application, Web-based email, or other network application. The persistence session remains in effect for a period of time, which you specify.

If a server participating in a persistence session goes DOWN, the load balancing virtual server uses the configured load balancing method to select a new service, and establishes a new persistence session with the server represented by that service. If the server goes OUT OF SERVICE, it continues to process existing persistence sessions, but the virtual server does not direct any new traffic to it. After the shutdown period elapses, the virtual server ceases to direct connections from existing clients to the service, closes existing connections, and redirects those clients to new services if necessary.

Depending on the persistence type you configure, the NetScaler appliance might examine the source IPs, destination IPs, SSL session IDs, Host or URL headers, or some combination of these things to place each connection in the proper persistence session. It might also base persistence on a cookie issued by the Web server, on an arbitrarily assigned token, or on a logical rule. Almost anything that allows the appliance to match connections with the proper persistence session and be used as the basis for persistence.

The following table summarizes the persistence types available on the NetScaler appliance.

Table 1. Types of Persistence

Persistence Type	Description
Source IP	SOURCEIP. Connections from the same client IP address are parts of the same persistence session.
HTTP Cookie	COOKIEINSERT. Connections that have the same HTTP Cookie header are parts of the same persistence session.
SSL Session ID	SSLSESSION. Connections that have the same SSL Session ID are parts of the same persistence session.
URL Passive	URLPASSIVE. Connections to the same URL are treated as parts of the same persistence session.
Custom Server ID	CUSTOMSERVERID. Connections with the same HTTP HOST header are treated as parts of the same persistence session.
Destination IP	DESTIP. Connections to the same destination IP are treated as parts of the same persistence session.
Source and	SRCIPDESTIP. Connections that are both from the same source IP and to the same destination IP

Persistence Type	Description
Destination IPs	are treated as parts of the same persistence session.
SIP Call ID	CALLID. Connections that have the same call ID in the SIP header are treated as parts of the same persistence session.
RTSP Session ID	RTSPSID. Connections that have the same RTSP Session ID are treated as parts of the same persistence session.
User-Defined Rule	RULE. Connections that match a user-defined rule are treated as parts of the same persistence session.

Depending on the type of persistence that you have configured, the virtual server can support either 250,000 simultaneous persistent connections or any number of persistent connections up to the limits imposed by the amount of RAM on your NetScaler appliance. The following table shows which types of persistence fall into each category.

Table 2. Persistence Types and Numbers of Simultaneous Connections Supported

Persistence Type	Number of Simultaneous Persistent Connections Supported
Source IP, SSL Session ID, Rule, destination IP, source IP/destination IP, SIP Call ID, RTSP Session ID	250 K
Cookie, URL Server ID, Custom Server ID	Memory limit. In case of CookieInsert, if timeout is not 0, the number of connections is limited by memory.

Some types of persistence are specific to particular types of virtual server. The following table lists each type of persistence and indicates which types of persistence are supported on which types of virtual server.

Table 3. Relationship of Persistence Type to Virtual Server Type

Persistence Type	HTTP	HTTPS	TCP	UDP/IP	SSL_Bridge	SSL_TCP	RTSP	SIP_UDP
SOURCEIP	YES	YES	YES	YES	YES	YES	NO	NO
COOKIEINSERT	YES	YES	NO	NO	NO	NO	NO	NO
SSLSESSION	NO	YES	NO	NO	YES	YES	NO	NO
URLPASSIVE	YES	YES	NO	NO	NO	NO	NO	NO

Persistence Type	HTTP	HTTPS	TCP	UDP/IP	SSL_Bridge	SSL_TCP	RTSP	SIP_UDP
CUSTOMSERVERID	YES	YES	NO	NO	NO	NO	NO	NO
RULE	YES	YES	YES	NO	NO		NO	NO
SRCIPDESTIP	YES	YES	YES	YES	YES	YES	NO	NO
DESTIP	YES	YES	YES	YES	YES	YES	NO	NO
CALLID	NO	NO	NO	NO	NO	NO	NO	YES
RTSPID	NO	NO	NO	NO	NO	NO	YES	NO

Persistence Based on Source IP Address

Mar 16, 2012

When source IP persistence is configured, the load balancing virtual server uses the configured load balancing method to select a service for the initial request, and then uses the source IP address (client IP address) to identify subsequent requests from that client and send them to the same service. You can set a time-out value, which specifies the maximum inactivity period for the session. When the time-out value expires, the session is discarded, and the configured load balancing algorithm is used to select a new server.

Caution: In some circumstances, using persistence based on source IP address can overload your servers. All requests to a single Web site or application are routed through the single gateway to the NetScaler appliance, even though they are then redirected to multiple locations. In multiple proxy environments, client requests frequently have different source IP addresses even when they are sent from the same client, resulting in rapid multiplication of persistence sessions where a single session should be created. This issue is called the "Mega Proxy problem." You can use HTTP cookie-based persistence instead of Source IP-based persistence to prevent this from happening.
To configure persistence based on Source IP Address, see Configuring Persistence Types That Do Not Require a Rule.

Note: If all incoming traffic comes from behind a Network Address Translation (NAT) device or proxy, the traffic appears to the NetScaler appliance to come from a single source IP address. This prevents Source IP persistence from functioning properly. Where this is the case, you must select a different persistence type.

Persistence Based on HTTP Cookies

Feb 11, 2015

When HTTP cookie persistence is configured, the NetScaler appliance sets a cookie in the HTTP headers of the initial client request. The cookie contains the IP address and port of the service selected by the load balancing algorithm. As with any HTTP connection, the client then includes that cookie with any subsequent requests.

When the NetScaler appliance detects the cookie, it forwards the request to the service IP and port in the cookie, maintaining persistence for the connection. You can use this type of persistence with virtual servers of type HTTP or HTTPS. This persistence type does not consume any NetScaler resources and therefore can accommodate an unlimited number of persistent clients.

Note: If the client's Web browser is configured to refuse cookies, HTTP cookie-based persistence will not work. It might be advisable to configure a cookie check on the Web site, and warn clients that do not appear to be storing cookies properly that they will need to enable cookies for the Web site if they want to use it.
The format of the cookie that the NetScaler appliance inserts is:

NSC_XXXX=<ServiceIP ><ServicePort>

where:

- NSC_XXXX is the virtual server ID that is derived from the virtual server name.
- ServiceIP and ServicePort are encoded representations of the service IP address and service port, respectively. The IP address and port are encoded separately.

You can set a time-out value for this type of persistence to specify an inactivity period for the session. When the connection has been inactive for the specified period of time, the NetScaler appliance discards the persistence session. Any subsequent connection from the same client results in a new server being selected based on the configured load balancing method, and a new persistence session being established.

Note: If you set the time-out value to 0, the NetScaler appliance does not specify an expiration time, but sets a session cookie that is not saved when the client's browser is shut down.
By default, the NetScaler appliance sets HTTP version 0 cookies for maximum compatibility with client browsers. (Only certain HTTP proxies understand version 1 cookies; most commonly used browsers do not.) You can configure the appliance to set HTTP version 1 cookies, for compliance with RFC2109. For HTTP version 0 cookies, the appliance inserts the cookie expiration date and time as an absolute Coordinated Universal Time (GMT). It calculates this value as the sum of the current GMT time on the appliance and the time-out value. For HTTP version 1 cookies, the appliance inserts a relative expiration time by setting the "Max-Age" attribute of the HTTP cookie. In this case, the client's browser calculates the actual expiration time.

To configure persistence based on a cookie inserted by the appliance, see Configuring Persistence Types That Do Not Require a Rule.

In the HTTP cookie, the appliance by default sets the httponly flag to indicate that the cookie is nonscriptable and should not be revealed to the client application. Therefore, a client-side script cannot access the cookie, and the client is not susceptible to cross-site scripting.

Certain browsers, however, do not support the httponly flag and, therefore, might not return the cookie. As a result, persistence is broken. For browsers that do not support the flag, you can omit the httponly flag in the persistence cookie.

To change the httponly flag setting by using the command line interface

At the command prompt, type:

set lb parameter -httpOnlyCookieFlag (ENABLED | DISABLED)

Example

```
>  set lb parameter -httpOnlyCookieFlag disabled
   Done
> show lb parameter
  Global LB parameters:
      Persistence Cookie HttpOnly Flag: DISABLED
      Use port for hash LB: YES
 Done
```

To change the httponly flag setting by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. In the Settings group, click Configure Load Balancing Parameters.
3. To not set the httponly flag in the persistence cookie, clear the Persistence Cookie HTTPOnly Flag check box.
4. Click OK.
5. Open the Configure Load Balancing Parameters dialog box and verify the setting you just configured.

Persistence Based on SSL Session IDs

Mar 16, 2012

When SSL Session ID persistence is configured, the NetScaler appliance uses the SSL Session ID, which is part of the SSL handshake process, to create a persistence session before the initial request is directed to a service. The load balancing virtual server directs subsequent requests that have the same SSL session ID to the same service. This type of persistence is used for SSL bridge services.

Note:
There are two issues that users should consider before choosing this type of persistence. First, the NetScaler appliance does not encrypt or decrypt data when it forwards requests to services in an SSL bridge configuration, because it must maintain the data structures to keep track of the sessions. This type of persistence therefore consumes resources on the NetScaler appliance, which limits the number of concurrent persistence sessions that it can support. If you expect to support a very large number of concurrent persistence sessions, you might want to choose another type of persistence.

Second, if the client and the load-balanced server should renegotiate the session ID during their transactions, persistence is not maintained, and a new persistence session is created when the client's next request is received. This may result in the client's activity on the Web site being interrupted and the client being required to reauthenticate or restart the session. It may also result in large numbers abandoned sessions if the timeout is set to too large a value.

To configure persistence based on SSL session ID, see Configuring Persistence Types That Do Not Require a Rule.

Persistence Based on Diameter AVP Number

Aug 29, 2013

You can use persistence based on the AVP number of a Diameter message to create persistent Diameter sessions. When the NetScaler appliance finds the AVP in the Diameter message, it creates a persistence session based on the value of the AVP. All subsequent messages that match the value of the AVP are directed to the previously selected server. If the value of the AVP does not match the persistence session, a new session is created for the new value.

Note: If the AVP number is not defined in Diameter base-protocol RFC 6733, and if the number is nested inside a grouped AVP, you must define a sequence of AVP numbers (maximum of 3) in parent-to-child order. For example, if persist AVP number X is nested inside AVP Y, which is nested in Z, define the list as Z Y X.

To configure Diameter-based persistence on a virtual server by using the command line interface

At the command prompt, type the following command:

set lb vserver <name> -PersistenceType <type-> persistAVPno <positive_integer>

Example

```
set lb vserver diameter_vs -persistenceType DIAMETER -persistAVPno 263
```

Custom Server ID Persistence

Sep 15, 2015

In the Custom Server ID persistence method, the Server ID specified in the client request is used to maintain persistence. For this type of persistence to work, you must first set a server ID on the services. The NetScaler appliance checks the URL of the client request and connects to the server associated with the specified server ID. The service provider should make sure that the users are aware of the server IDs to be provided in their requests for specific services.

For example, if your site provides different types of data, such as images, text, and multimedia, from different servers, you can assign each server a server ID. On the NetScaler appliance, you specify those server IDs for the corresponding services, and you configure custom server ID persistence on the corresponding load balancing virtual server. When sending a request, the client inserts the server ID into the URL indicating the required type of data.

To configure custom server ID persistence:

- In your load balancing setup, assign a server ID to each service for which you want to use the user-defined server ID to maintain persistence. Alphanumeric server IDs are allowed.
- Specify rules, in the default-syntax expression language, to examine the URL queries for the server ID and forward traffic to the corresponding server.
- Configure custom server ID persistence.

Note: The persistence time-out value does not affect the Custom Server ID persistence type. There is no limit on the maximum number of persistent clients because this persistence type does not store any client information.

Example

In a load balancing setup with two services, assign server ID 2345-photo-56789 to Service-1, and server ID 2345-drawing-abb123 to Service-2. Bind these services to a virtual server named Web11.

```
set service Service-1 10.102.29.5 -CustomServerID 2345-photo-56789
set service Service-2 10.102.29.6 -CustomServerID 2345-drawing-abb123
```

On virtual server Web11, enable Custom Server ID persistence.

Example

```
set lb vserver Web11 -persistenceType customserverID
bind lb vserver Web11 Service-[1-2]
```

Create the following expression so that all URL queries containing the string "sid=" are examined.

```
HTTP.REQ.URL.AFTER_STR("sid=")
```

When a client sends a request with the following URL to the IP address of Web11, the NetScaler directs the request to Service-2 and honors persistence.

Example

```
http://www.example.com/index.asp?&sid=2345-drawing-abb123
```

For more information about default-syntax policy expressions, see the Policy Configuration and Reference.

To configure custom server ID persistence by using the configuration utility

1. Assign a server ID to each of the services for which you want to configure custom server ID persistence.
   1. Navigate to Traffic Management > Load Balancing > Services.
   2. In the details pane, select the service for which you want to specify a server ID, and then click Open.

3. In the Configure Service dialog box, click the Advanced tab.
4. Scroll down, and under Others, in the Server ID box, type an ID for the server.
5. Click OK.

2. Configure custom server ID persistence on the virtual server to which the services are bound.
    1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
    2. In the details pane, select the virtual server for which you want to specify persistence, and then click Open.
    3. On the Method and Persistence tab, in the Persistence group, select CUSTOMSERVERID.
    4. In the Rule box, type an expression, or click Configure, and use the options available in the Create Expression dialog box, to create the expression.
    5. Click OK.

Persistence Based on IP Addresses

Jun 03, 2015

You can base persistence on Destination IP addresses, or on both Source IP and Destination IP Addresses.

Persistence Based on Destination IP Addresses

Updated: 2013-09-24

With destination IP address-based persistence, when the NetScaler appliance receives a request from a new client, it creates a persistence session based on the IP address of the service selected by the virtual server (the destination IP address). Subsequently, it directs requests to the same destination IP to the same service. This type of persistence is used with link load balancing. For more information about link load balancing, see Link Load Balancing.

The time-out value for destination IP persistence is the same as that for source IP persistence, described in Persistence Based on Source IP Address.

To configure persistence based on the destination IP address, see Configuring Persistence Types That Do Not Require a Rule.

Persistence Based on Source and Destination IP Addresses

With source and destination IP address-based persistence, when the NetScaler appliance receives a request, it creates a persistence session based on both the IP address of the client (the source IP address) and the IP address of the service selected by the virtual server (the destination IP address). Subsequently, it directs requests from the same source IP and to the same destination IP to the same service.

The time-out value for destination IP persistence is the same as that for source IP persistence, described in Persistence Based on Source IP Address.

To configure persistence based on both source and destination IP addresses, see Configuring Persistence Types That Do Not Require a Rule.

Persistence Based on SIP Call ID

Mar 16, 2012

With SIP Call ID persistence, the NetScaler appliance chooses a service based on the call ID in the SIP header. This enables it to direct packets for a particular SIP session to the same service and, therefore, to the same load balanced server. This persistence type is applicable specifically to SIP load balancing. For more information about SIP load balancing, see Monitoring SIP Services.

To configure persistence based on SIP Call ID, see Configuring Persistence Types That Do Not Require a Rule.

Persistence Based on RTSP Session IDs

Aug 29, 2013

With RTSP Session ID persistence, when the NetScaler appliance receives a request from a new client, it creates a new persistence session based on the Real-Time Streaming Protocol (RTSP) session ID in the RTSP packet header, and then directs the request to the RTSP service selected by the configured load balancing method. It directs subsequent requests that contain the same session ID to the same service. This persistence type is applicable specifically to SIP load balancing. For more information about SIP load balancing, see Monitoring SIP Services.

Note: RTSP Session ID persistence is configured by default on RTSP virtual servers, and you cannot modify that setting. Sometimes different RTSP servers issue the same session IDs. When this happens, unique sessions cannot be created between the client and the RTSP server by using only the RTSP session ID. If you have multiple RTSP servers that may issue the same session IDs, you can configure the appliance to append the server IP address and port to the session ID, creating a unique token that can be used to establish persistence. This is called session ID mapping.

To configure persistence based on RTSP Session IDs, see Configuring Persistence Types That Do Not Require a Rule.

Important: If you need to use session ID mapping, you must set the following parameter when configuring each service within the load balancing setup. Also, make sure that no non-persistent connections are routed through the RTSP virtual server.

Configuring URL Passive Persistence

Nov 23, 2015

With URL Passive persistence, when the NetScaler appliance receives a request from a client, it extracts the server IP address-port information (expressed as a single hexadecimal number) from the client request.

URL passive persistence requires configuring an advanced expression that specifies the query element that contains the server IP address-port information. For more information about classic and advanced policy expressions, see Policy Configuration and Reference.

The following expression configures the appliance to examine requests for URL queries that contain the string "urlp=", extract the server IP address-port information, convert it from a hexadecimal string to an IP and port number, and forward the request to the service configured with this IP address and port number.

HTTP.REQ.URL.AFTER_STR("urlp=")

If URL passive persistence is enabled and the above expression is configured, a request with the following URL and server IP address-port string is directed to 10.102.29.10:80.

http://www.example.com/index.asp?&urlp=0A661D0A0050

The persistence time-out value does not affect this persistence type; persistence is maintained as long as the server IP address-port information can be extracted from client requests. This persistence type does not consume any NetScaler resources, so it can accommodate an unlimited number of persistent clients.

To configure URL passive persistence, you first configure persistence as described in Configuring Persistence Types That Do Not Require a Rule. You set the persistence type to URLPASSIVE. You then perform the procedures provided below.

To configure URL passive persistence by using the command line interface

At the command prompt, type:

set lb vserver <vserverName> [-persistenceType <persistenceType>] [-rule <expression>]

Example

set lb vserver LB-VServer-1 -persistenceType URLPASSIVE –rule HTTP.REQ.URL.AFTER_STR("urlp=")

To configure URL passive persistence by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Configure Virtual Server (Load Balancing) dialog box, on the Method and Persistence tab, in Persistence, select URLPASSIVE.
3. Click the Configure button next to the Rule field.
4. In the dialog box that appears, create the rule that you want to use. For more information about creating rules, see Policy Configuration and Reference.
5. Click OK.

Configuring Persistence Based on User-Defined Rules

Aug 29, 2013

When rule based persistence is configured, the NetScaler appliance creates a persistence session based on the contents of the matched rule before directing the request to the service selected by the configured load balancing method. Subsequently, it directs all requests that match the rule to the same service. You can configure rule based persistence for services of type HTTP, SSL, RADIUS, ANY, TCP, and SSL_TCP.

Rule based persistence requires a classic or default syntax expression. You can use a classic expression to evaluate request headers, or you can use a default syntax expression to evaluate request headers, Web form data in a request, response headers, or response bodies. For example, you could use a classic expression to configure persistence based on the contents of the HTTP Host header. You could also use a default syntax expression to configure persistence based on application session information in a response cookie or custom header. For more information on creating and using classic and default syntax expressions, see Policy Configuration and Reference.

The expressions that you can configure depends on the type of service for which you are configuring rule based persistence. For example, certain RADIUS-specific expressions are not allowed for protocols other than RADIUS, and TCP-option based expressions are not allowed for service types other than the ANY type. For TCP and SSL_TCP service types, you can use expressions that evaluate TCP/IP protocol data, Layer 2 data, TCP options, and TCP payloads.

Note: For a use case that involves configuring rule based persistence on the basis of Financial Information eXchange ("FIX") Protocol data transmitted over TCP, see Configuring Rule Based Persistence Based on a Name-Value Pair in a TCP Byte Stream.
Rule based persistence can be used for maintaining persistence with entities such as Branch Repeater appliances, Branch Repeater plug-ins, cache servers, and application servers.

Note: On an ANY virtual server, you cannot configure rule-based persistence for the responses.
To configure persistence based on a user-defined rule, you first configure persistence as described in Configuring Persistence Types That Do Not Require a Rule, and set the persistence type to RULE. You then perform the procedures provided below. You can configure rule based persistence by using the configuration utility or the NetScaler command line.

To configure persistence based on user-defined rules by using the command line interface

At the command prompt, type:

set lb vserver <vserverName> [-rule <expression>][-resRule <expression>]
Example

```
set lb vserver vsvr_name –rule http.req.header("cookie").value(0).typecast_nvlist_t('=',';').value("server")

set lb vserver vsvr_name –resrule http.res.header("set-cookie").value(0).typecast_nvlist_t('=',';').value("server")
```

To configure persistence based on user-defined rules by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click **Add**.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Method and Persistence tab, choose the type of rule you want to configure.
   - If you want to base the rule on the request, click the Configure button next to the Rule field.
   - If you want to base the rule on the response, click the Configure button next to the Response Rule field.
4. In the dialog box that appears, select Switch to Classic Syntax or Switch to Advanced Syntax.
5. Select or create the rule that you want to use. Some examples of rules that you might find useful are provided below. For more information, see Policy Configuration and Reference.
6. Click OK.

Example: Classic Expression for a Request Payload

The following classic expression creates a persistence session based on the presence of a User-Agent HTTP header that contains the string, “MyBrowser”, and directs any subsequent client requests that contain this header and string to the same server that was selected for the initial request.

```
http header User-Agent contains MyBrowser
```

Example: Default syntax Expression for a Request Header

The following default syntax expression does exactly the same thing as the previous classic expression.

```
HTTP.REQ.HEADER("User-Agent").CONTAINS ("MyBrowser")
```

Example: Default syntax Expression for a Response Cookie

The following expression examines responses for “server” cookies, and then directs any requests that contain that cookie to the same server that was selected for the initial request.

```
HTTP.RES.HEADER("SET-COOKIE").VALUE(0).TYPECAST_NVLIST_T('=',';').VALUE("server")
```

Configuring Persistence Types That Do Not Require a Rule

Aug 29, 2013

To configure persistence, you must first set up a load balancing virtual server, as described in Setting Up Basic Load Balancing. You then configure persistence on the virtual server.

To configure persistence on a virtual server by using the command line interface

At the command prompt, type the following commands to configure persistence and verify the configuration:

- set lb vserver <name> -PersistenceType <type> [-timeout <integer>]
- show lb vserver

Example

```
set lb vserver Vserver-LB-1 -persistenceType SOURCEIP

show lb vserver
```

Note: For IP-based persistence, you can also set the persistMask parameter.

To configure persistence on a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure persistence, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Method and Persistence tab, in the Persistence list, select the persistence type you want to use (for example, SOURCEIP).
4. In the Time-out and Netmask text boxes type the time-out and subnet mask values (for example, 2 and 255.255.255.255).
5. Click OK.

Configuring Backup Persistence

Nov 12, 2013

The NetScaler appliance uses backup persistence to choose a new type of persistence when the primary persistence type fails. For example, if the primary persistence type is set to Cookie Insert, and backup persistence is set to Source IP, the NetScaler appliance uses Source IP-based persistence when the cookie is missing from the HTTP header or when the client browser does not support cookies.

You can set a time-out value for backup persistence only when the primary persistence type is HTTP Cookie-based or RTSP session ID-based persistence, and the backup persistence type is Source IP-based.

To set backup persistence for a virtual server by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -persistenceType <PersistenceType> -persistenceBackup <BackupPersistenceType>

Example

set lb vserver Vserver-LB-1 -persistenceType CookieInsert -persistenceBackup SourceIP

To set backup persistence for a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure backup persistence (for example, Vserver-LB-1), and then click Open.
3. The Configure Virtual Server (Load Balancing) dialog box, click the Method and Persistence tab.
4. In the Persistence list, select the persistence type you want (for example, COOKIEINSERT).
5. In the Time-out text box, type the time-out value you want (for example, 20).
6. In the Backup Persistence list, select the backup persistence type that you want to configure (for example, SOURCEIP).
7. In the Backup Time-out and Netmask text boxes, type the backup time-out value and netmask (for example, 20 and 255.255.255.255).
8. Click OK.

Configuring Persistence Groups

Aug 29, 2013

When you have load-balanced servers that handle several different types of connections (such as Web servers that host multimedia), you can configure a virtual server group to handle these connections. To create a virtual server group, you bind different types of virtual servers, one for each type of connection that your load balanced servers accept, into a single group. You then configure a persistence type for the entire group.

You can configure either source IP-based persistence or HTTP cookie-based persistence for persistence groups. After you set persistence for the entire group, you cannot change it for individual virtual servers in the group. If you configure persistence on a group and then add a new virtual server to the group, the persistence of the new virtual server is changed to match the persistence setting of the group.

When persistence is configured on a group of virtual servers, persistence sessions are created for initial requests, and subsequent requests are directed to the same service as initial request, regardless of the virtual server in the group that receives each client request.

If you configure HTTP cookie-based persistence, the domain attribute of the HTTP cookie is set. This setting causes the client software to add the HTTP cookie into client requests if different virtual servers have different public host names. For more information about CookieInsert persistence type, see Persistence Based on HTTP Cookies.

To create a virtual server persistency group by using the command line interface

At the command prompt, type:

bind lb group <vServerGroupName> <vServerName> -persistenceType <PersistenceType>

Example

bind lb group Vserver-Group-1 Vserver-LB-1 -persistenceType CookieInsert

To create a virtual server persistency group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Persistency Groups.
2. On the Persistency Groups pane, click Add.
3. In the Create Persistency Group dialog box, in the Group Name text box type a name for the group (for example, Vserver-Group-1).
4. In the Persistence list, select a persistence type (for example, SOURCEIP).
5. In the Persistence Mask and Time-out text boxes, type the persistence mask and timeout values (for example, 255.255.255.255 and 2).
6. Under Virtual Server List, in the Available Virtual Server list box, select the virtual server that you want to bind to the group (for example, Vserver-LB-1), and then click Add.
7. Click Create, and then click Close. The virtual server group you created appears in the Persistence Groups pane.

You can also change the backup persistence type, backup persistence time-out, and cookie domain value on an existing persistence group.

To modify a virtual server group by using the command line interface

At the command prompt, type:

set lb group <vServerGroupName> -PersistenceBackup <BackupPersistenceType> -persistMask <SubnetMaskAddress>

Example

```
set lb group vserver-Group-1 -PersistenceBackup SourceIP -persistMask 255.255.255.255
```

To modify a virtual server group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Persistency Groups.
2. In the Persistency Groups pane, select the virtual server group that you want to modify (for example, Vserver-Group-1), and click Open.
3. The Configure Virtual Server Group dialog box appears.
4. In the Backup Persistence list, select the type of backup persistence (for example, SOURCEIP).
5. In the Persistence Mask text box, type the subnet mask (for example, 255.255.255.255).
6. Click OK.

Configuring RADIUS Load Balancing with Persistence

Jun 08, 2015

Today's complex networking environment often requires coordinating a high-volume, high-capacity load balancing configuration with robust authentication and authorization. Application users may connect to a VPN through mobile access points such as consumer-grade DSL or Cable connections, WiFi, or even dial-up nodes. Those connections usually use dynamic IPs, which can change during the connection.

If you configure RADIUS load balancing on the NetScaler appliance to support persistent client connections to RADIUS authentication servers, the appliance uses the user logon or the specified RADIUS attribute instead of the client IP as the session ID, directing all connections and records associated with that user session to the same RADIUS server. Users are therefore able to log on to your VPN from mobile access locations without experiencing disconnections when the client IP or WiFi access point changes.

To configure RADIUS load balancing with persistence, you must first configure RADIUS authentication for your VPN. For information and instructions, see the Authentication, Authorization, Auditing (AAA) chapter in AAA Application Traffic. You must also choose either the Load Balancing or Content Switching feature as the basis for your configuration, and make sure that the feature you chose is enabled. The configuration process with either feature is almost the same.

Then, you configure either two load balancing, or two content switching, virtual servers, one to handle RADIUS authentication traffic and the other to handle RADIUS accounting traffic. Next, you configure two services, one for each load balancing virtual server, and bind each load balancing virtual server to its service. Finally, you create a load balancing persistency group and set the persistency type to RULE.

To configure RADIUS load balancing with persistence, see the following sections:

- Enabling the Load Balancing or Content Switching Feature
- Configuring Virtual Servers
- Configuring Services
- Binding Virtual Servers to Services
- Configuring a Persistency Group for Radius

Enabling the Load Balancing or Content Switching Feature

Updated: 2013-08-29

To use the Load Balancing or Content Switching feature, you must first ensure that the feature is enabled. If you are configuring a new NetScaler appliance that has not previously been configured, both of these features are already enabled, so you can skip to the next section. If you are configuring a NetScaler appliance with a previous configuration on it, and you are not certain that the feature you will use is enabled, you must do that now.

- For instructions on enabling the load balancing feature, see Enabling Load Balancing.
- For instructions on enabling the content switching feature, see Enabling Content Switching.

Configuring Virtual Servers

Updated: 2013-09-13

After enabling the load balancing or content switching feature, you must next configure two virtual servers to support RADIUS authentication:

- **RADIUS authentication virtual server.** This virtual server and its associated service will handle authentication traffic to your RADIUS server. Authentication traffic consists of connections associated with users logging onto your protected application or virtual private network (VPN).
- **RADIUS accounting virtual server.** This virtual server and its associated service will handle accounting connections to your RADIUS server. Accounting traffic consists of connections that track an authenticated user's activities on your protected application or VPN.

Important: You must create either a pair of load balancing virtual servers or a pair of content switching virtual servers to use in your RADIUS persistence configuration. You cannot mix virtual server types.

To configure a load balancing virtual server by using the command line interface

At the command prompt type the following commands to create a new load balancing virtual server and verify the configuration:

- add lb vserver <name> RADIUS <IP address> <port> -lbmethod TOKEN -rule <rule>
- show lb vserver <name>

To configure an existing load balancing virtual server, replace the above add lb virtual server command with the set lb vserver command, which takes the same arguments.

To configure a content switching virtual server by using the command line interface

At the command prompt type the following commands to create a new content switching virtual server and verify the configuration:

- add cs vserver <name> RADIUS <IP address> <port> -lbmethod TOKEN -rule <rule>
- show cs vserver <name>

To configure an existing content switching virtual server, replace the above add cs vserver command with the set cs vserver command, which takes the same arguments.

Example

```
add lb vserver radius_auth_vs1 RADIUS 192.168.46.33 1812
    -lbmethod TOKEN -rule CLIENT.UDP.RADIUS.USERNAME
add lb vserver radius_acct_vs1 RADIUS 192.168.46.34 1813
    -lbmethod TOKEN -rule CLIENT.UDP.RADIUS.USERNAME
set lb vserver radius_auth_vs1 RADIUS 192.168.46.33 1812
    -lbmethod TOKEN -rule CLIENT.UDP.RADIUS.USERNAME
set lb vserver radius_auth_vs1 RADIUS 192.168.46.34 1813
    -lbmethod TOKEN -rule CLIENT.UDP.RADIUS.USERNAME
```

To configure a load balancing or content switching virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers or Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, do one of the following:

- To create a new virtual server, click Add.
- To modify an existing virtual server, select the virtual server, and then click Open.

3. In the Create Virtual Server (Load balancing) or Configure Virtual Server (Content Switching) dialog box, specify values for the following parameters:
   - Name*—name
   - Protocol*—protocol
   - IP address*—IPAddress
   - Port*—port

   * A required parameter

4. In the Method and Persistence tab, specify values for the following parameters:
   - Method*—method
   - Rule*—rule

   * A required parameter

5. Click Close. The virtual server that you created now appears in the Virtual Servers pane.

Configuring Services

Updated: 2013-09-17

After configuring your virtual servers, you must next configure two services, one for each of the virtual servers that you created. For instructions, see Configuring Services.

Note: Once configured, these services are in the DISABLED state until the NetScaler appliance can connect to your RADIUS server's authentication and accounting IPs and monitor their status.

Binding Virtual Servers to Services

After configuring your services, you must next bind each of the virtual servers that you created to the appropriate service. For instructions, see Binding Services to the Virtual Server.

Configuring a Persistency Group for Radius

Updated: 2013-09-17

After binding your load balancing virtual servers to the corresponding services, you must set up your RADIUS load balancing configuration to support persistence. To do so, you configure a load balancing persistency group that contains your RADIUS load balancing virtual servers and services, and configure that load balancing persistency group to use rule-based persistence. For instructions, see Configuring Persistence Groups.

Viewing Persistence Sessions

Sep 13, 2013

You can view the different persistence sessions that are in effect globally or for a particular virtual server.

Note: A NetScaler nCore appliance uses multiple CPU cores for packet handling. Every session on the appliance is owned by a CPU core. If the appliance receives a request for which a session does not already exist, a session is created, and one of the cores is designated as the owner of that session. Subsequent requests that belong to that session might not always arrive at and be handled by the owner core. In that case, inter-core messaging ensures that the session information on the owner core is always current. However, when a core receives a request that belongs to a persistence session owned by another core, the inter-core messaging does not refresh the timeout value for the persistence session. Consequently, in the output of successively executed show lb persistentSessions commands, which display timeout values from owner cores only, the timeout value for a persistence session might diminish to 0 (zero), even if the persistence session continues to be active.

To view a persistence session by using the command line interface

At the command prompt, to view all persistence sessions type:

show lb persistentSessions [<vServer>]

Example

show lb persistentSessions myVserver

To view persistence sessions by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. In the details pane, under Monitor Sessions, click Virtual Server persistence sessions.

Clearing Persistence Sessions

Nov 12, 2013

You might need to clear persistence sessions from the NetScaler if sessions fail to time out. You can do one of the following:

- Clear all sessions for all virtual servers at once.
- Clear all sessions for a given virtual server at once.
- Clear a particular session that is associated with a given virtual server.

Note: The functionality for clearing a particular session that is associated with a given virtual server is available only on NetScaler 10.e.

To clear a persistence session by using the command line interface

At the command prompt, type the following commands to clear persistence sessions and verify the configuration:

- clear lb persistentSessions [<vServer> [-persistenceParam <string>]]
- show persistentSessions <vServer>

Examples

Example 1 clears all persistence sessions for load balancing virtual server lbvip1. Example 2 first displays the persistence sessions for load balancing virtual server lbvip1, clears the session with persistence parameter xls, and then displays the persistence sessions to verify that the session was cleared.

Example

```
> clear persistentSessions lbvip1
 Done
> show persistentSessions
 Done
>
```

Example 2

```
> show persistentSessions lbvip1
Type      SRC-IP   ...   PERSISTENCE-PARAMETER
RULE      0.0.0.0  ...   xls
RULE      0.0.0.0  ...   txt
RULE      0.0.0.0  ...   html
 Done
> clear persistentSessions lbvip1 -persistenceParam xls
 Done
> show persistentSessions lbvip1
Type      SRC-IP   ...   PERSISTENCE-PARAMETER
RULE      0.0.0.0  ...   txt
RULE      0.0.0.0  ...   html
 Done
>
```

To clear persistence sessions by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. In the details pane, under Monitor Sessions, click Clear persistence sessions.

3. In the Clear Persistence Sessions dialog box, do one of the following:
   - If you want to clear all sessions for all virtual servers on the appliance, in Virtual Server, select All Virtual Servers.
   - If you want to clear all sessions for a given virtual server, in Virtual Server, select the virtual server.
   - If you want to clear a particular session, in Virtual Server, select the virtual server, and then, in Persistence Parameter, select the persistence parameter whose session you want to clear.
4. Click OK.

Overriding Persistence Settings for Overloaded Services

Aug 29, 2013

When a service is loaded or is otherwise unavailable, service to clients is degraded. To work around this situation, you might have to configure the NetScaler appliance to temporarily forward to other services the requests that would otherwise be included in the persistence session that is associated with the overloaded service. In other words, you might have to override the persistence setting that is configured for the load balancing virtual server. You can achieve this functionality by setting the skippersistency parameter. With the parameter set, when the virtual server receives new connections for an overloaded service, the virtual server disregards any existing persistence sessions that are associated with that service, until the service returns to a state at which it can accept requests. Persistence sessions associated with other services are not affected. The functionality is available for only virtual servers whose type is ANY or UDP.

In Branch Repeater load balancing configurations, you must also configure a load monitor and bind it to the service. The monitor takes the service out of subsequent load balancing decisions until the load on the service is brought below the configured threshold. For information about configuring a load monitor for your virtual server, see Understanding Load Monitors.

You can configure the virtual server to perform one of the following actions with the requests that would otherwise form a part of the persistence session:

- **Send each request to one of the other services.** The virtual server takes a load balancing decision and sends each request to one of the other services on the basis of the configured load balancing method. If all the services are overloaded, requests are dropped until a service becomes available.
  Both wildcard and IP address–based virtual servers support this option. This action is appropriate for all deployments, including deployments in which the virtual server is load balancing Branch Repeater appliances or firewalls.

- **Bypass the virtual server-service configuration.** The virtual server does not take a load balancing decision. Instead, it simply bridges each request through to a physical server on the basis of the destination IP address in the request.
  Only wildcard virtual servers of type ANY and UDP support the bypass option. Wildcard virtual servers have a *:* IP and port combination. This action is appropriate for deployments in which you are using the virtual server to load balance Branch Repeater appliances or firewalls. In these deployments, the NetScaler appliance first forwards a request to a Branch Repeater appliance or firewall, and then forwards the processed response to a physical server. If you configure the virtual server to bypass the virtual server–service configuration for overloaded services, if a Branch Repeater appliance or firewall gets overloaded, the virtual server bridges requests directly to their destination IP addresses until the Branch Repeater appliance or firewall can accept requests.

To override persistence settings for overloaded services by using the command line interface

At the command prompt, type the following commands to override persistence settings for overloaded services and verify the configuration:

- set lb vserver <name> -skippersistency <skippersistency>
- show lb vserver <name>

Example

```
> set lb vserver mylbvserver -skippersistency ReLb
```

```
 Done
> show lb vserver mylbvserver
 mylbvserver (*:*) - ANY Type: ADDRESS
   . . .
   . . .
 Skip Persistency: ReLb
   . . .
 Done
>
```

To override persistence settings for overloaded services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click the virtual server that you want to override persistence settings when services are overloaded, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, in the Skip Persistency list, select the option that you want.
4. Click OK.

Troubleshooting

Feb 28, 2014

The statistics from the NetScaler VPX appliance indicate that the appliance has reached the session persistence limit. As a result, persistence sessions are failing. Is possible to increase the session persistence limit?

Cause: The NetScaler appliance has the system limit of 250,000 persistence session for a core.

Resolution: To resolve this issue, you can perform any of the following tasks:

- Reduce the time out value for persistence
- Increase the number of cores for the appliance

After configuring Cookie Insert persistence on the NetScaler appliance, the users report that the connections work fine for some time, but then start getting disconnected. What best practice should I follow when configuring persistence?

Cause: By default, the time-out value for Cookie Insert persistence is 120 seconds.

Resolution: When you configure persistence for applications for which idle time cannot be determined, set the Cookie Insert persistence time-out value to 0. With this setting, the connection does not time out.

After configuring an HTTP virtual server on the NetScaler appliance, I need to make sure that a user always connects to the same server for the requested content, so I configured SourceIP persistence. Now, increasing the time-out value for persistence introduces latency. How can I increase the timeout value without affecting performance?

Resolution: Consider using Cookie Insert persistence with the time-out value set to 0. This setting enables long-duration persistence settings, because the appliance does not specify a time for expiring the cookie.

After configuring Cookie Insert persistence on the NetScaler appliance, it works as expected when clients from the same time zone access the content. However, when a client from another time zone makes an attempt to connect, the connection is immediately timed out.

Cause: Time based Cookie Insert persistence works as expected when a client from the same time zone makes a connection. However, when the client machine and NetScaler appliance are in different time zones, the cookie is not valid. For example, when a client in EST time zone sends a cookie at 11:00 AM EST to a NetScaler appliance in the PST time zone, the appliance receives the cookie at 2:00 PM PST. As a result of the difference in time, the cookie is not valid, and the connection is immediately timed-out.

Resolution: Set the time-out value for Cookie Insert persistence to 0.

A NetScaler appliance is used to load balance application servers, such as Oracle Weblogic server. To make sure that clients get persistent connections to these servers, SourceIP persistence is configured. It works as expected when a connection is made from a computer. However, when thin clients attempt a connection through a terminal server and, as a result, the appliance receives requests from multiple clients from the same IP address (the terminal server IP address). Therefore, the connections from all thin clients are directed to the same application server. Is it possible to configure persistency for requests from individual thin clients based on the client IP address?

Cause: The NetScaler appliance receives requests from the terminal server and the source IP address of the request remains the same. As a result, the appliance cannot distinguish among the requests received from the thin clients and provide persistence according to the requests from thin clients.

Resolution: To avoid this problem, you can configure Rule persistence based on some unique parameter value for each thin client.

The NetScaler appliance is used to load balance Web Interface servers. When accessing the servers, the user receives the "State Error" error message. Additionally, when one of the Web Interface servers is shut down or not available, some of the users receive an error message.

Cause: Lack of persistence to the Web Interface servers can result in error messages when a user attempts to connect to the server.

Resolution: Citrix recommends that you specify the Cookie Insert persistence method on the NetScaler appliance when load balancing Web Interface servers.

Customizing a Load Balancing Configuration

Jun 09, 2015

After you configure a basic load balancing setup, you can make a number of modifications to it so that it distributes load exactly as you need. The load balancing feature is complex. You can modify the basic elements by changing the load balancing algorithm, configuring load balancing groups and using them to create your load balancing configuration, configuring persistent client-server connections, configuring the redirection mode, and assigning different weights to different services that have different capacities.

The default load balancing algorithm on the NetScaler appliance is the least connection method, which configures the appliance to send each incoming connection to the service that is currently handling the fewest connections. You can specify different load balancing algorithms, each of which is suited to different conditions.

To accommodate applications such as shopping carts, which require that all requests from the same user be directed to the same server, you can configure the appliance to maintain persistent connections between clients and servers. You can also specify persistence for a group of virtual servers, causing the appliance to direct individual client requests to the same service regardless of which virtual server in the group receives the client request.

You can enable and configure the redirection mode that the appliance uses when redirecting user requests, choosing between IP-based and MAC-based forwarding. You can assign weights to different services, specifying what percentage of incoming load should be directed to each service, so that you can include servers with different capacities in the same load balancing setup without overloading the lower-capacity servers or allowing the higher-capacity servers to sit idle.

This section includes the following details:

- Customizing the Hash Algorithm for Persistence across Virtual Servers
- Configuring the Redirection Mode
- Configuring per-VLAN Wildcarded Virtual Servers
- Assigning Weights to Services
- Configuring the MySQL and Microsoft SQL Server Version Setting

Customizing the Hash Algorithm for Persistence across Virtual Servers

Aug 29, 2013

The NetScaler appliance uses hash-based algorithms for maintaining persistence across virtual servers. By default, the hash-based load balancing method uses a hash value of the IP address and port number of the service. If a service is made available at different ports on the same server, the algorithm generates different hash values. Therefore, different load balancing virtual servers might send requests for the same application to different services, breaking the pseudo-persistence.

As an alternative to using the port number to generate the hash value, you can specify a unique hash identifier for each service. For a service, the same hash identifier value must be specified on all the virtual servers. If a physical server serves more than one type of application, each application type should have a unique hash identifier.

The algorithm for computing the hash value for a service works as follows:

- By default, a global setting specifies the use of port number in a hash calculation.
- If you configure a hash identifier for a service, it is used, and the port number is not, regardless of the global setting.
- If you do not configure a hash identifier, but change the default value of the global setting so that it does not specify use of the port number, the hash value is based only on the IP address of the service.
- If you do not configure a hash identifier or change the default value of the global setting to use the port number, the hash value is based on the IP address and the port number of the service.

You can also specify hash identifiers when using the NetScaler command line to bind services to a service group. In the configuration utility, you can open a service group and add hash identifiers on the Members tab.

To change the use-port-number global setting by using the command line interface

At the command prompt, type:

set lb parameter -usePortForHashLb (YES | NO)

Example

```
> set lb parameter -usePortForHashLb NO
 Done
>show lb parameter
Global LB parameters:
      Persistence Cookie HttpOnly Flag: DISABLED
      Use port for hash LB: NO
 Done
```

To change the use-port-number global setting by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. In the Settings group, click Configure Load Balancing Parameters.
3. To not use the port number to generate the hash value, clear the Use Port for Hash Based LB Methods check box.
4. Click OK.
5. Open the Configure Load Balancing Parameters dialog box and verify the setting you just configured.

To create a new service and specify a hash identifier for a service by using the command line interface

At the command prompt, type the following commands to set the hash ID and verify the setting:

add service < name > (< ip > |< serverName >) < serviceType > < port > -hashId < positive_integer >

show service <name>

Example

```
> add service flbkng 10.101.10.1 http 80 -hashId 12345
 Done
>show service flbkng
      flbkng (10.101.10.1:80) - HTTP
      State: DOWN
      Last state change was at Thu Nov  4 10:14:52 2010
      Time since last state change: 0 days, 00:00:15.990
      Server Name: 10.101.10.1
      Server ID : 0   Monitor Threshold : 0

      Down state flush: ENABLED
      Hash Id: 12345

1)     Monitor Name: tcp-default
            State: DOWN     Weight: 1

 Done
```

To specify a hash identifier for an existing service by using the command line interface

Type the set service command, the name of the service, and **-hashID** followed by the ID value.

To specify a hash identifier while adding a service group member

To specify a hash identifier for each member to be added to the group and verify the setting, at the command prompt, type the following commands (Be sure to specify a unique hashID for each member.):

bind servicegroup <serviceGroupName> <memberName> <port> -hashId <positive_integer>

show servicegroup <serviceGroupName>

Example

```
 bind servicegroup http_svc_group 10.102.27.153 80 -hashId 2222222

>show servicegroup SRV
      SRV - HTTP
      State: ENABLED  Monitor Threshold : 0
      ...

      1)       1.1.1.1:80   State: DOWN     Server Name: 1.1.1.1    Server ID: 123  Weight: 1
      Hash Id: 32211
```



```
        Monitor Name: tcp-default      State: DOWN
    ...

    2)      2.2.2.2:80   State: DOWN    Server Name: 2.2.2.2    Server ID: 123  Weight: 1
    Hash Id: 12345

        Monitor Name: tcp-default      State: DOWN
    ...
Done
```

To specify a hash identifier for a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, do one of the following:
   - To create a new service, click Add.
   - To modify an existing service, select the service and then click Open.
3. In the Create Service or Configure Service dialog box:
   - ServiceName*—name
   - Server*—ip or serverName
   - Protocol*—serviceType
   - Port*—port
4. Click the Advanced tab and then scroll down in the dialog box.
5. In the Hash ID box, type a unique hash ID value.
6. Click Create.
7. Open the service and verify the settings you just configured.

To specify a hash identifier for an already configured service group member by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. Select a service group, and then click Open.
3. On the Members tab, under Configured Members, in the row of the member for which you want to specify a hash ID value, double-click the space in the Hash ID column.
4. Type a unique hash ID value.
5. Click OK.
6. Open the service group and verify the hash IDs of the service group members you just configured.

Configuring the Redirection Mode

Aug 02, 2016

The redirection mode configures the method used by a virtual server to determine where to forward incoming traffic. The NetScaler appliance supports the following redirection modes:

- IP-Based forwarding (the default)
- MAC-Based forwarding

You can configure MAC-Based forwarding on networks that use direct server return (DSR) topology, link load balancing, or firewall load balancing. For more information on MAC-Based forwarding, see Configuring MAC-Based Forwarding.

To configure the redirection mode by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -m <RedirectionMode>

Example

set lb vserver Vserver-LB-1 -m MAC

To configure the redirection mode by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the redirection mode, and then click Open.
3. On the Advanced tab, under Redirection Mode, click either IP-Based or MAC-Based.
4. Click OK.

Configuring per-VLAN Wildcarded Virtual Servers

Nov 12, 2013

If you want to configure load balancing for traffic on a specific virtual local area network (VLAN), you can create a wildcarded virtual server with a listen policy that restricts it to processing traffic only on the specified VLAN.

To configure a wildcarded virtual server that listens to a specific VLAN by using the command line interface

At the command prompt, type the following commands to configure a wildcarded virtual server that listens to a specific VLAN and verify the configuration:

- add lb vserver <name> <serviceType> IPAddress * Port * -listenpolicy <expression> [-listenpriority <positive_integer>]
- show vserver

Example

```
add lb vserver Vserver-LB-vlan1 ANY -listenpolicy "CLIENT.VLAN.ID.EQ(2)" -listenpriority 10

show vserver Vserver-LB-vlan1
```

To configure a wildcarded virtual server that listens to a specific VLAN by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, do one of the following:
   - To create a new virtual server, click Add.
   - To modify an existing virtual server, select the virtual server, and then click Open.
3. In the Create Virtual Server or Configure Virtual Server dialog box, on the Services tab, specify values for the following parameters:
   - Name*—name
   - Protocol*—serviceType
   - IP address*—IPAddress
   - Port—port

   *A required parameter
4. In the Advanced tab, expand Listen Policy, and then specify values for the following parameters:
   - Listen Priority*—listenpriority
   - Listen Policy Rule*—rule

   *A required parameter
5. Click Create or OK, depending on whether you are creating a new virtual server or modifying an existing virtual server.
6. Click Close. The virtual server that you created now appears in the Virtual Servers page.
7. To remove a virtual server, in the Virtual Servers pane select the virtual server, and then click Remove.

After you have created this virtual server, you bind it to one or more services as described in Setting Up Basic Load Balancing.

Assigning Weights to Services

Nov 12, 2013

In a load balancing configuration, you assign weights to services to indicate the percentage of traffic that should be sent to each service. Services with higher weights can handle more requests; services with lower weights can handle fewer requests. Assigning weights to services allows the NetScaler appliance to determine how much traffic each load balanced server can handle, and therefore more effectively balance load.

Note: If you use a load balancing method that supports weighting of services (for example, the round robin method), you can assign a weight to the service.

The following table describes the load balancing methods that support weighting, and briefly describes the manner in which weighting affects how a service is selected for each one.

Load Balancing Methods	**Service Selection with Weights**
Round Robin	The virtual server prioritizes the queue of available services such that services with the highest weights come to the front of the queue more frequently than those with the lowest weights and receive proportionately more traffic. For a complete description, see The Round Robin Method.
Least Connection	The virtual server selects the service with the best combination of fewest active transactions and highest weight. For a complete description, see The Least Connection Method.
Least Response Time and Least Response Time Method using Monitors	The virtual server selects the service with the best combination of fewest active transactions and fastest average response time. For a complete description, see The Least Response Time Method.
Least Bandwidth	The virtual server selects the service with the best combination of least traffic and highest bandwidth. For a complete description, see The Least Bandwidth Method.
Least Packets	The virtual server selects the service with the best combination of fewest packets and highest weight. For a complete description, see The Least Packets Method.
Custom Load	The virtual server selects the service with the best combination of lowest load and highest weight. For a complete description, see The Custom Load Method.
Hashing methods and Token method	Weighting is not supported by these load balancing methods.

To configure a virtual server to assign weights to services by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -weight <Value> <ServiceName>

Example

set lb vserver Vserver-LB-1 -weight 10 Service-HTTP-1

To configure a virtual server to assign weights to services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server, and then click Open.
3. On the Services tab, in the Weights spin box, type or select the weight to assign to the service (for example, 10).
4. Click OK.

Configuring the MySQL and Microsoft SQL Server Version Setting

Mar 24, 2015

You can specify the version of Microsoft® SQL Server® and the MySQL server for a load balancing virtual server that is of type MSSQL and MySQL respectively. The version setting is recommended if you expect some clients to not be running the same version as your MySQL or Microsoft SQL Server product. The version setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

At the command prompt, type the following commands to set the Microsoft SQL Server version parameter for a load balancing virtual server and verify the configuration:

- set lb vserver <name> -mssqlServerVersion <mssqlServerVersion>
- show lb vserver <name>

Example

```
> set lb vserver myMSSQLvip -mssqlServerVersion 2008R2
 Done
> show lb vserver myMSSQLvip
 myMSSQLvip (190.0.2.12:1433) - MSSQL Type: ADDRESS
. . .
. . .
 MSsql Server Version: 2008R2
. . .
. . .
 Done
>
```

At the command prompt, type the following commands to set the MySQL Server version parameter for a load balancing virtual server and verify the configuration:

- set lb vserver <name> -mysqlServerVersion <string>
- show lb vserver <name>

Example

```
> set lb vserver mysqlsvr -mysqlserverversion 5.5.30
 Done
> sh lb vserver mysqlsvr
  mysqlsvr (2.22.2.222:3306) - MYSQL      Type: ADDRESS
. . .
. . .
 Mysql Server Version: 5.5.30
. . .
```

```
. . .
 Done
>
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the setting, and then click Open.
3. In the Configure Virtual Server dialog box, do the following:
    1. In the advanced tab, click MsSql or MySql.
    2. In the Server Version list, select the version of your Microsoft SQL Server product.
    3. Click Create or OK, and then click Close.

Configuring Diameter Load Balancing

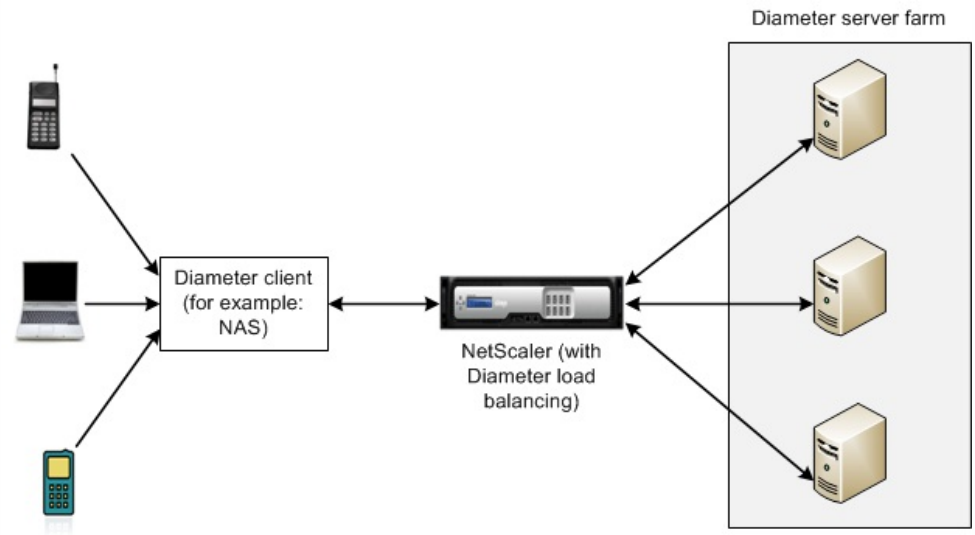
Mar 18, 2013

The Diameter protocol is a next generation Authentication, Authorization, and Accounting (AAA) signaling protocol used mainly on mobile devices such as laptops and mobile phones. It is a peer-to-peer protocol, as opposed to the traditional client-server model used by most other protocols. However, in most Diameter deployments, the clients originates the request and the server responds to the request.

When Diameter messages are exchanged, the Diameter server usually does much more processing than does the Diameter client. With the increase in control plane signaling volume, the Diameter server becomes a bottleneck. Therefore, Diameter messages must be load balanced to multiple servers. A virtual server performing load balancing of Diameter messages provides the following benefits:

- Lighter load on Diameter servers, which translates to faster response time to end users.
- Server health monitoring and better failover capabilities.
- Better scalability in terms of server addition without changing client configuration.
- High availability.
- SSL-Diameter offloading.

The following figure shows a Diameter system in a NetScaler deployment:



A Diameter system has the following components:

- **Diameter client.** Supports Diameter client applications in addition to the base protocol. Diameter clients are often implemented in devices situated at the edge of a network and provide access control services for that network. Typical examples of Diameter clients are a Network Access Server (NAS) and the Mobile IP Foreign Agent (FA).
- **Diameter agent.** Provides relay, proxy, redirect, or translation services. The NetScaler appliance (configured with a Diameter load balancing virtual server) plays the role of a Diameter agent.
- **Diameter server.** Handles the authentication, authorization, and accounting requests for a particular realm. A Diameter server must support Diameter server applications in addition to the base protocol.

In a typical Diameter topology, when an end-user device (such as a mobile phone) needs a service, it sends a request to a Diameter client. Each Diameter client establishes a single connection (TCP connection—SCTP is not yet supported) with a Diameter server as specified by the Diameter base-protocol RFC 6733. The connection is long-lived and all messages between

the two Diameter nodes (client and server) are exchanged over this connection. The NetScaler uses message based load balancing .

Example

A mobile service provider uses Diameter for its billing system. When a subscriber uses a prepaid number, the Diameter client repeatedly sends requests to the server to check the available balance. The Diameter protocol establishes a connection between the client and the server, and all requests are exchanged over that connection. Connection based load balancing would be pointless, because there is only one connection. However, with the large number of messages on the connection, message based load balancing expedites the process of billing the prepaid mobile subscriber.

A Diameter client opens a connection to the NetScaler appliance and sends a Diameter capability exchange (CER) message. Diameter messages are composed of command codes and each command has a set of Attribute-Value Pairs (AVPs), such as Origin-Host and Host-IP-Address.

The NetScaler selects a Diameter server, opens a connection to the server, and forwards the CER message to the server. The server reads the client identity and determines that it is directly connected to the client.

The Diameter server prepares the Diameter handshake reply and sends it to the NetScaler appliance. The appliance modifies the handshake and inserts its own identity. At this point, the Diameter client determines that it is directly connected to the NetScaler (the agent).

Note: Until the Diameter handshake is complete, all Diameter request messages from the client are queued on the selected server. The packets are forwarded to the server when the handshake is complete.

Load Balancing Diameter Traffic

When a client sends a request to the NetScaler appliance, the appliance parses the request and contextually load balances it to a Diameter server on the basis of a persist AVP. The NetScaler has advertised the client identity to the server, so it does not add route entries, because the server is expecting messages directly from client.

Server initiated requests are not as frequent as client requests. Server initiated requests are similar to client initiated requests, except:

- Since messages are received from multiple servers, the NetScaler maintains the transaction state by adding a unique Hop by Hop (HbyH) number to each forwarded request message. When the message response arrives (with same HbyH number), the appliance translates this HbyH number to the HbyH number that was received on the server when the request arrived.
- NetScaler adds a route entry by putting its identity, because the client sees the NetScaler as a relay agent.

Note: If a Diameter message spans more than one packet, the NetScaler accumulates the packets in an incomplete header queue and forwards them to the server when the full message is accumulated. Similarly, if a single packet contains more than one Diameter message, the NetScaler splits the packet and forwards the messages to servers as determined by the load balancing virtual server.

Disconnecting a Session

A Disconnect Peer Request (DPR) indicates the peer's intention of closing the connection, with the reason for closing the connection. The peer replies with a DPA (TCP always provides successful DPA).

- When the NetScaler receives a DPR from the client, it broadcasts the DPR to all servers and immediately replies with a DPA to the client. The servers reply with DPAs, but the NetScaler ignores them. The client sends a FIN, which the NetScaler broadcasts to all servers.
- When the NetScaler receives a DPR from the server, it replies with a DPA to that server alone, and does not remove the

server from the reuse pool. When the server sends a FIN, the NetScaler replies with FIN/ACK and removes connections from the reuse pool.

- If the NetScaler receives a FIN from the client, it sends the client a FIN/ACK, broadcasts the FIN, and immediately removes the server connection from the reuse pool.
- If the NetScaler receives a FIN from the server, it sends a FIN/ACK and removes it from reuse pool. Any new message for this server is sent on a new connection.

Updated: 2013-11-12

To configure the NetScaler appliance to load balance Diameter traffic, you must first set the Diameter parameters on the appliance, then add the Diameter monitor, add the Diameter services, bind the services to the monitor, add the Diameter load balancing virtual server, and bind the services to the virtual server.

To configure load balancing for Diameter traffic by using the command line interface

1. Configure the Diameter parameters.
   set ns diameter -identity <string> -realm <string> -serverClosePropagation <YES | NO>

   Example
   set ns diameter -identity mydomain.org -realm org -serverClosePropagation YES
2. Add a Diameter monitor.
   add lb monitor <monitorName> DIAMETER -originHost <string> -originRealm <string>

   Example
   add lb monitor diameter_mon DIAMETER -originHost mydomain.org -originRealm org
3. Create the Diameter services.
   add service <name>@ <IP>@ DIAMETER <port>

   Example
   add service diameter_svc0 10.102.82.86 DIAMETER 3868
   add service diameter_svc1 10.102.82.87 DIAMETER 3868
   add service diameter_svc2 10.102.82.88 DIAMETER 3868
   add service diameter_svc3 10.102.82.89 DIAMETER 3868
4. Bind the Diameter services to the Diameter monitor.
   bind service <name>@ monitorName <monitorName>

   -Example
   bind service diameter_svc0 -monitorName diameter_mon
   bind service diameter_svc1 -monitorName diameter_mon
   bind service diameter_svc2 -monitorName diameter_mon
   bind service diameter_svc3 -monitorName diameter_mon
5. Add a Diameter load balancing virtual server with Diameter persistence.
   add lb vserver <name>@ DIAMETER <IPAddress> <port> -persistenceType DIAMETER -persistAVPno <positive_integer>

   Example
   add lb vserver diameter_vs DIAMETER 10.102.112.152 3868 -persistenceType DIAMETER -persistAVPno 263
6. Bind the Diameter services to the Diameter load balancing virtual server.
   bind lb vserver <name>@ <serviceName>

   Example

```
bind lb vserver diameter_vs diameter_svc0
bind lb vserver diameter_vs diameter_svc1
bind lb vserver diameter_vs diameter_svc2
bind lb vserver diameter_vs diameter_svc3
```

7. Save the configuration.
   save ns config

   Note: You can also configure load balancing of Diameter traffic over SSL by using the **SSL_DIAMETER** service type.

To configure load balancing for Diameter traffic by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, click Change Diameter Parameters.
3. In the Configure Diameter Parameters dialog box, specify values for the following parameters. (Hover on a parameter's name to display its description.)
   - Host Identity
   - Realm
   - Server Close Propagation
4. Click OK to configure the Diameter parameters.
5. Navigate to Traffic Management > Load Balancing > Monitors.
6. In the details pane, click Add.
7. In the Create Monitor dialog box, create a monitor with Type as Diameter.
8. On the Special Parameters tab, enter values for Origin Host and Origin Realm.
9. Navigate to Traffic Management > Load Balancing > Services.
10. In the details pane, click Add.
11. In the Create Service dialog box, create a service with Protocol as Diameter and bind the created Diameter monitor to this service.
12. Navigate to Traffic Management > Load Balancing > Virtual Servers.
13. In the details pane, click Add.
14. In the Create Virtual Server (Load Balancing) dialog box, create a load balancing virtual server with Protocol as DIAMETER and bind the created Diameter services to this virtual server.
15. On the Method and Persistence tab, select the Persistence as DIAMETER and specify a **AVP Number**.

Protecting a Load Balancing Configuration against Failure

Jun 03, 2015

When a load balancing virtual server fails, or when the virtual server is unable to handle excessive traffic, the load balancing setup can fail. You can protect your load balancing setup against failure by configuring the NetScaler appliance to redirect excess traffic to an alternate URL, configuring a backup load balancing virtual server, and configuring stateful connection failover.

To protect a load balancing configuration against failure, see the following sections:

- Redirecting Client Requests to an Alternate URL
- Configuring a Backup Load Balancing Virtual Server
- Configuring Spillover
- Connection Failover
- Flushing the Surge Queue

Redirecting Client Requests to an Alternate URL

Sep 02, 2013

In the event that a load balancing virtual server of type HTTP or type HTTPS goes DOWN or is disabled, you can redirect requests to an alternate URL by using an HTTP 302 redirect. The alternate URL can provide information about the status of the server.

You can redirect to a page on the local server or a remote server. You can redirect to a relative URL or an absolute URL. If you configure a redirect to a relative URL consisting of a domain name with no path, the NetScaler appliance appends the path of the incoming URL to the domain. If you use an absolute URL, the HTTP redirect is sent to that URL with no modification.

Note: If a load balancing virtual server is configured with both a backup virtual server and a redirect URL, the backup virtual server takes precedence over the redirect URL. A redirect is used only when both the primary and backup virtual servers are DOWN.

At the command prompt, type:

set lb vserver <vServerName> -redirectURL <URLValue>

Example

set lb vserver Vserver-LB-1 -redirectURL http://www.newdomain.com/mysite/maintenance

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure redirect URL, and then click Open.
3. On the Advanced tab, in the Redirect URL text box, type the URL (for example, http://www.newdomain.com/mysite/maintenance).
4. Click OK.

Configuring a Backup Load Balancing Virtual Server

Sep 02, 2013

You can configure the NetScaler appliance to direct requests to a backup virtual server in the event that the primary load balancing virtual server is DOWN or unavailable. The backup virtual server is a proxy and is transparent to the client. The appliance can also send a notification message to the client regarding the site outage.

You can configure a backup load balancing virtual server when you create it, or you can change the optional parameters of an existing virtual server. You can also configure a backup virtual server for an existing backup virtual server, thus creating cascading backup virtual servers. The maximum depth of cascading backup virtual servers is 10.

If you have multiple virtual servers that connect to two servers, you have a choice for what happens if the primary virtual server goes DOWN and then comes back up. The default behavior is for the primary virtual server to resume its role as primary. However, you may want to configure the backup virtual server to remain in control in the event that it takes over. For example, you may want to sync updates on the backup virtual server to the primary virtual server and then manually force the original primary server to resume its role. In this case, you can designate the backup virtual server to remain in control in the event that the primary virtual server goes DOWN and then comes back up.

You can configure a redirect URL on the primary load balancing virtual server as a fallback for when both the primary and the backup virtual servers are DOWN or have reached their threshold for handling requests. When services bound to virtual servers are OUT OF SERVICE, the appliance uses the redirect URL.

Note: If a load balancing virtual server is configured with both a backup virtual server and a redirect URL, the backup virtual server takes precedence over the redirect URL. A redirect is used only when the primary and backup virtual servers are down.

At the command prompt, type:

set lb vserver <vServerName> -backupVserver <BackupVServerName> [-disablePrimaryOnDown]

Example

```
set lb vserver Vserver-LB-1 -backupVserver Vserver-LB-2 -disablePrimaryOnDown
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the backup virtual server, and then click Open.
3. On the Advanced tab, in the Backup Virtual Server list, select the backup virtual server.
4. If you want the backup virtual server to remain in control until you manually re-enable the primary virtual server even if the primary virtual server comes back up, select the Disable Primary When Down check box.
5. Click OK.

Configuring Spillover

Sep 02, 2013

A spillover configuration on the appliance consists of a primary virtual server that is configured with a spillover method, a spillover threshold, and a backup virtual server. Backup virtual servers can also be configured for spillover, creating a chain of backup virtual servers.

The spillover method specifies the operational condition on which you want to base your spillover configuration (for example, the number of established connections, bandwidth, or combined health of the server farm). When a new connection arrives, the appliance verifies that the primary virtual server is up and compares the operational condition with the configured spillover threshold. If the threshold is reached, the spillover feature diverts new connections to the first available virtual server in the backup chain. The backup virtual server manages the connections it receives until the load on the primary falls below the threshold.

If you configure spillover persistence, the backup virtual server continues to process the connections it received, even after the load on the primary falls below the threshold. If you configure spillover persistence and a spillover persistence timeout, the backup virtual server processes connections for only the specified period of time after the load on the primary falls below the threshold.

Note: In most cases, spillover is triggered if the value associated with the spillover method exceeds the threshold (for example, number of connections). Keep in mind, however, that with the server-health spillover method, spillover is triggered if the health of the server farm falls below the threshold.
You can configure spillover in one of the following ways:

- Specify a predefined spillover method. Four predefined methods are available, and they fulfill common spillover requirements.
- Configure policy based spillover. In policy based spillover, you use a NetScaler rule to specify the conditions that should be met for spillover to occur. NetScaler rules give you the flexibility to configure spillover for various operational conditions.

Use policy based spillover if a predefined method does not satisfy your requirements. If you configure both for a primary virtual server, the policy based spillover configuration takes precedence over the predefined method.

First, you create the primary virtual server and the virtual servers that you need for the backup chain. You set up the backup chain by specifying one virtual server as the backup for the primary (that is, you create a secondary virtual server), a virtual server as the backup for the secondary (that is, you create a tertiary virtual server), and so on. Then, you configure spillover by either specifying a predefined spillover method or creating and binding spillover policies.

For instructions for assigning a virtual server as the backup for another virtual server, see Configuring a Backup Load Balancing Virtual Server.

Updated: 2013-09-02

Predefined spillover methods fulfill some of the more common spillover requirements. To use one of the predefined spillover methods, you configure spillover parameters on the primary virtual server. To create a chain of backup virtual servers, you also configure spillover parameters on backup virtual servers.

If the backup virtual servers reach their own threshold values, and the service type is TCP, the NetScaler appliance sends clients a TCP reset. For service types HTTP, SSL, and RTSP , it diverts new requests to the redirect URL configured for the primary virtual server. A redirect URL can be specified for only HTTP, SSL, and RTSP virtual servers. If a redirect URL is not configured, the NetScaler appliance sends clients a TCP reset (if the virtual server is of type TCP) or an HTTP 503 response (if the virtual server is of type HTTP or SSL).

Note: With RTSP virtual servers, the NetScaler appliance uses only data connections for spillover. If the backup RTSP virtual server is not available, the requests are redirected to an RTSP URL and an RTSP redirect message is sent to the client.

To configure a predefined spillover method for a virtual server by using the command line interface

At the command prompt, type:

set lb vserver <vServerName> -soMethod <spillOverType> -soThreshold <positiveInteger> -soPersistence ENABLED -soPersistenceTimeout <positiveInteger>

```
set lb vserver Vserver-LB-1 -soMethod Connection -soThreshold 1000 -soPersistence enabled -soPersistenceTimeout 2
```

To configure a predefined spillover method for a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure the spillover, and then click Open.
3. On the Advanced tab, in the Method list, select the type of threshold, and in Threshold text box, type the threshold value.
4. Under Spillover, select the Persistence check box, and in Persistence Time-out (min) text box type the time-out.
5. Click OK.

Updated: 2013-09-02

Spillover policies, which are based on rules (expressions), enable you to configure the appliance for a wider range of spillover scenarios. For example, you can configure spillover on the basis of the virtual server's response time, or on the basis of number of connections in the virtual server's surge queue.

To configure policy based spillover, first create a spillover action. You then select the expression that you want to use in the spillover policy , configure the policy, and associate the action with it. Finally, you bind the spillover policy to a load balancing, content switching, or global server load balancing virtual server. You can bind multiple spillover policies to a virtual server, with priority numbers. The appliance evaluates the spillover policies in ascending order of priority numbers and performs the action associated with the last policy to evaluate to TRUE.

A virtual server can also have a backup action. The backup action is performed if the virtual server does not have one or more backup virtual servers, or if all of the backup virtual servers are DOWN, disabled, or have reached their own spillover limits.

When a spillover policy results in an UNDEF condition (an exception thrown when the result of policy evaluation is undefined), an UNDEF action is performed. The UNDEF action is always ACCEPT. You cannot specify an UNDEF action of your choice.

Configuring a Spillover Action

Updated: 2013-09-13

A spillover action is performed when the spillover policy with which it is associated evaluates to TRUE. Currently, SPILLOVER is the only supported spillover action.

At the command prompt, type the following commands to configure a spillover policy and verify the configuration:

- add spillover action <name> -action SPILLOVER
- show spillover action <name>

Example

```
> add spillover action mySoAction -action SPILLOVER
 Done
> show spillover action mySoAction
1) Name: mySoAction Action: SPILLOVER
 Done
>
```

Selecting an Expression for the Spillover Policy

Updated: 2013-09-02

In the policy expression, you can use any virtual-server based expression that returns a Boolean value. For example, you could use one of the following expressions:

SYS.VSERVER("vserver").RESPTIME.GT(<int>), SYS.VSERVER("vserver").STATE.EQ("<string>"), and SYS.VSERVER("vserver").THROUGHPUT.LT(<int>).

In addition to the existing functions such as RESPTIME, STATE, and THROUGHPUT, you can use the following virtual server based functions that have been introduced with this feature:

Averagesurgecount

Returns the average number of requests in the surge queues of active services. Returns 0 (zero) if there are no active services. Raises an UNDEF condition if used with a content switching or global server load balancing virtual server.

Activeservices

Returns the number of active services. Raises an UNDEF condition if used with a content switching or global server load balancing virtual server.

Activetransactions

Returns the value of the virtual-server-level counter for current active transactions.

is_dynamic_limit_reached

Returns a Boolean TRUE if the number of connections being managed by the virtual server equals the dynamically calculated threshold. The dynamic threshold is the sum of the maximum client (Max Clients) settings of the bound services that are UP.

You can use a policy expression to implement any of the predefined spillover methods. The following table maps the predefined spillover methods to the expressions you can use to implement them:

Table 1. Converting predefined spillover methods to policy expressions

Predefined spillover method	Corresponding expression
CONNECTION	SYS.VSERVER(\"<vserver-name>\").CONNECTIONS, used with the GT(int) arithmetic function.
BANDWIDTH	SYS.VSERVER(\"<vserver-name>\").THROUGHPUT, used with the GT(int) arithmetic function.
HEALTH	SYS.VSERVER(\"<vserver-name>\").HEALTH, used with the LT(int) arithmetic function.
DYNAMICCONNECTION	SYS.VSERVER(\"<vserver-name>\").IS_DYNAMIC_LIMIT_REACHED Note: If you implement policy based spillover by using the IS_DYNAMIC_LIMIT_REACHED function, you must also configure the predefined DYNAMICCONNECTION method for the virtual server, so that statistics required for spillover to work are collected.

Configuring a Spillover Policy

Updated: 2013-09-13

A spillover policy uses a Boolean expression as a rule to specify the conditions that must be met for spillover to occur.

At the command prompt, type the following commands to configure a spillover policy and verify the configuration:

- add spillover policy <name> -rule <expression> -action <string> [-comment <string>]
- show spillover policy <name>

Example

```
> add spillover policy mySoPolicy -rule SYS.VSERVER("v1").RESPTIME.GT(50) -action mySoAction -comment "Triggers spillover when the vserver's response time is greater
 Done
> show spillover policy mySoPolicy
1) Name: mySoPolicy Rule: "SYS.VSERVER(\"v1\").RESPTIME.GT(50)" Action: mySoAction Hits: 0 ActivePolicy: 0
 Comment: "Triggers spillover when the vserver\'s response time is greater than 50 ms."
 Done
>
```

Binding a Spillover Policy to a Virtual Server

Updated: 2013-09-13

You can bind a spillover policy to load balancing, content switching, or global server load balancing virtual servers). You can bind multiple policies to a virtual server, with Goto expressions controlling the flow of evaluation.

At the command prompt, type the following commands to bind a spillover policy to a load balancing, content switching, or global server load balancing virtual server and verify the configuration:

- bind (lb | cs | gslb) vserver <name> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>]
- show (lb | cs | gslb) vserver <name>

Example

```
> bind lb vserver vserver1 -policyName mySoPolicy -priority 5
 Done
> show lb vserver vserver1
 vserver1 (2.2.2.12:80) - HTTP Type: ADDRESS
 . . .

1) Spillover Policy Name: mySoPolicy Priority: 5
 GotoPriority Expression: END
 Flowtype: REQUEST
 Done
>
```

Configuring a Backup Action for a Spillover Event

Updated: 2013-09-02

A backup action specifies what to do in the event that the spillover threshold is reached but one or more backup virtual serves are either not configured or are down, disabled, or have reached their own thresholds.

Note: For the predefined spillover methods that are configured directly on the virtual server (as values of the Spillover Method parameter), the backup action is not configurable. By default, the appliance sends clients a TCP reset (if the virtual server is of type TCP) or an HTTP 503 response (if the virtual server is of type HTTP or SSL).
The backup action is configured on the virtual server. You can configure the virtual server to accept requests (after the threshold specified by the policy is reached), redirect clients to a URL, or simply drop requests until the number of requests falls below the threshold.

At the command prompt, type the following commands to configure a backup action and verify the configuration:

- set lb vserver <name> -soBackupAction <soBackupAction>
- show lb vserver <name>

Example

```
> set lb vserver vs1 -soBackupAction REDIRECT -redirectURL http://www.mysite.com/maintenance
 Done
> show lb vserver vs1
 vs1 (10.102.29.76:80) - HTTP Type: ADDRESS
 State: UP
 . . .
 Redirect URL: http://www.mysite.com/maintenance
          . . .
Done
>
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. Select the virtual server for which you want to configure a backup spillover action, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. In the Spillover area of the tab, in Backup Action, select an action.
5. Click OK.

Connection Failover

Sep 30, 2013

Connection failover helps prevent disruption of access to applications deployed in a distributed environment. In a NetScaler High Availability (HA) setup, *connection failover* (or *connection mirroring-CM*) refers to keeping active an established TCP or UDP connection when a failover occurs. The new primary NetScaler appliance has information about the connections established before the failover and continues to serve those connections. After failover, the client remains connected to the same physical server. The new primary appliance synchronizes the information with the new secondary appliance by using the SSF framework. If the L2Conn parameter is set, Layer 2 connection parameters are also synchronized with the secondary.

You can set up connection failover in either stateless or stateful mode. In the stateless connection failover mode, the HA nodes do not exchange any information about the connections that are failed over. This method has no runtime overhead.

In the stateful connection failover mode, the primary appliance synchronizes the data of the failed-over connections with the new secondary appliance.

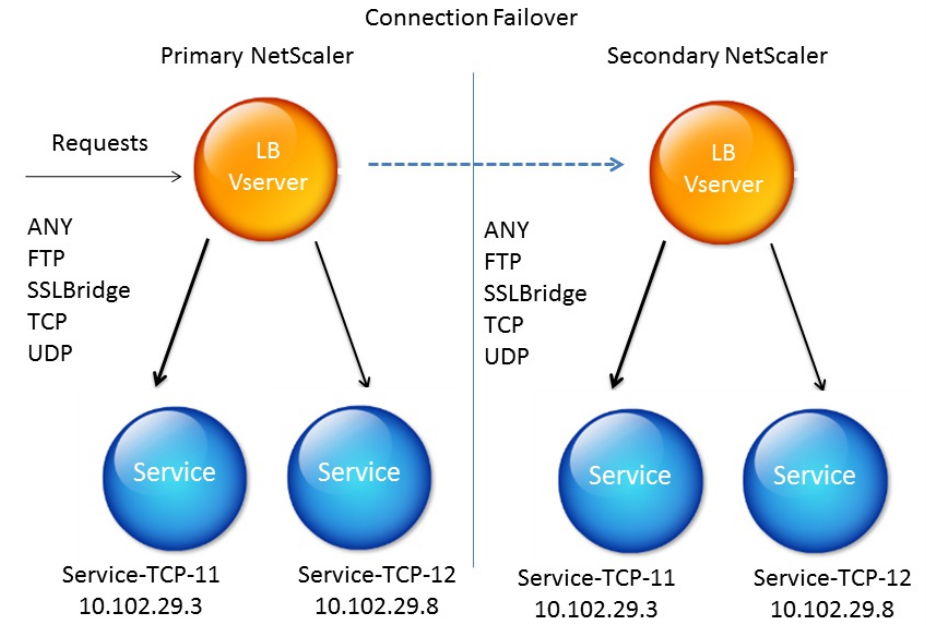
In stateless connection failover, the new primary appliance tries to re-create the packet flow according to the information contained in the packets it receives.

In stateful failover, to maintain current information about the mirrored connections, the primary appliance sends messages to the secondary appliance. The secondary appliance maintains the data related to the packets but uses it only in the event of a failover. If a failover occurs, the new primary (old secondary) appliance starts using the stored data about the mirrored connections and accepting traffic. During the transition period, the client and server may experience a brief disruption and retransmissions.

Note:
Verify that the primary appliance is able to authorize itself on the secondary appliance. To verify correct configuration of the passwords, use the show rpcnode command from command line or use the RPC option of the Network menu from the configuration utility.

A basic HA configuration with connection failover contains the entities shown in the following figure.

Figure 1. Connection Failover Entity Diagram



Connection failover can be configured only on load balancing virtual servers. It cannot be configured on content-switching virtual servers.

The following table describes the setup supported for connection failover.

Table 1. Connection Failover - Supported Setup

Setting	Stateless	Stateful
Service type	ANY.	ANY, UDP, TCP, FTP, SSL_BRIDGE.
Load balancing methods	All methods supported for the service type ANY. However, if Source IP persistence is not set, the SRCIPSRCPORTHASH method must be used.	All methods applicable to the supported service types.
Persistence types	SOURCEIP persistence.	All types applicable to the supported service types are supported.
USIP	Must be ON.	No restriction. It can be ON or OFF.
Service bindings	Service can be bound to only one virtual server.	Service can be bound to one or more virtual servers.
Internet	IPv4 and IPv6	IPV4

Setting	Stateless	Stateful
Protocol (IP) versions		
Redundancy support	Clustering and high availability	High availability

The following table lists the features affected if connection failover is configured.

Table 2. How Connection Failover Affects NetScaler Features

Feature	Impact of Connection Failover
SYN protection	For any connection, if a failover occurs after the NetScaler issues SYN-ACK but before it receives the final ACK, the connection is not supported by connection failover. The client must reissue the request to establish the connection.
Surge protection	If the failover occurs before a connection with the server is established, the new primary NetScaler tries to establish the connection with the server. It also retransmits all the packets held in the course of surge protection.
Access down	If enabled, the access-down functionality takes precedence over connection failover.
Application Firewall™	The Application Firewall feature is not supported.
INC	Independent network configuration is not supported in the high availability (HA) mode.
TCP buffering	TCP buffering is not compatible with connection mirroring.
Close on response	After failover, the NATPCBs may not be closed on response.
IPv6 virtual servers	Not yet supported.

Configuring Connection Failover

Sep 13, 2013

You can configure connection failover on a load balancing virtual server.

At the command prompt, type:

set lb vserver <vServerName> -connFailover <Value>

Example

```
set lb vserver Vserver-LB-1 -connFailover stateful
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers pane, select the virtual server for which you want to configure connection failover, and click Open.
3. On the Advanced tab, in the Connection Failover drop-down list, select Stateful.
4. Click OK.

Disabling Connection Failover

Sep 02, 2013

When connection failover is disabled on a virtual server, the resources allocated to the virtual server are freed.

At the command prompt, type:

set lb vserver <vServerName> -connFailover <Value>

Example

```
set lb vserver Vserver-LB-1 -connFailover disable
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers pane, select the virtual server for which you want to configure a connection failover and click Open.
3. On the Advanced tab, in the Connection Failover drop-down list box, select Disable.
4. Click OK.

Flushing the Surge Queue

Dec 04, 2013

When a physical server receives a surge of requests, it becomes slow to respond to the clients that are currently connected to it, which leaves users dissatisfied and disgruntled. Often, the overload also causes clients to receive error pages. To avoid such overloads, the NetScaler appliance provides features such as surge protection, which controls the rate at which new connections to a service can be established.

The appliance does connection multiplexing between clients and physical servers. When it receives a client request to access a service on a server, the appliance looks for an already established connection to the server that is free. If it finds a free connection, it uses that connection to establish a virtual link between the client and the server. If it does not find an existing free connection, the appliance establishes a new connection with the server, and establishes a virtual link between client and the server. However, if the appliance cannot establish a new connection with the server, it sends the client request to a surge queue. If all the physical servers bound to the load balancing or content switching virtual server reach the upper limit on client connections (max client value, surge protection threshold or maximum capacity of the service), the appliance cannot establish a connection with any server. The surge protection feature uses the surge queue to regulate the speed at which connections are opened with the physical servers. The appliance maintains a different surge queue for each service bound to the virtual server.

The length of a surge queue increases whenever a request comes for which the appliance cannot establish a connection, and the length decreases whenever a request in the queue gets sent to the server or a request gets timed out and is removed from the queue.

If the surge queue for a service or service group becomes too long, you may want to flush it. You can flush the surge queue of a specific service or service group, or of all the services and service groups bound to a load balancing virtual server. Flushing a surge queue does not affect the existing connections. Only the requests present in the surge queue get deleted. For those requests, the client has to make a fresh request.

You can also flush the surge queue of a content switching virtual server. If a content switching virtual server forwards some requests to a particular load balancing virtual server, and the load balancing virtual server also receives some other requests, when you flush the surge queue of the content switching virtual server, only the requests received from this content switching virtual server are flushed; the other requests in the surge queue of the load balancing virtual server are not flushed.

Note: You cannot flush the surge queues of cache redirection, authentication, VPN or GSLB virtual servers or GSLB services.
Note: Do not use the Surge Protection feature if Use Source IP (USIP) is enabled.

The flush ns surgeQ command works in the following manner:

- You can specify the name of a service, service group, or virtual server whose surge queue has to be flushed.
- If you specify a name while executing the command, surge queue of the specified entity will be flushed. If more than one entity has the same name, the appliance flushes surge queues of all those entities.
- If you specify the name of a service group, and a server name and port while executing the command, the appliance flushes the surge queue of only the specified service group member.
- You cannot directly specify a service group member (<serverName> and <port>) without specifying the name of the service group (<name>) and you cannot specify <port> without a <serverName>. Specify the <serverName> and <port> if you want to flush the surge queue for a specific service group member.
- If you execute the command without specifying any names, the appliance flushes the surge queues of all the entities present on the appliance.
- If a service group member is identified with a server name, you must specify the server name in this command; you cannot specify its IP address.

At the command prompt, type:

flush ns surgeQ [-name <name>] [-serverName <serverName> <port>]

Examples

1.
flush ns surgeQ –name SVC1ANZGB –serverName 10.10.10.1 80
The above command flushes the surge queue of the service or virtual server that is named SVC1ANZGB and has IP address as 10.10.10
2.
flush ns surgeQ
The above command flushes all the surge queues on the appliance.

1. Navigate to Traffic Management > Load Balancing.
2. To select an entity, do one of the following:
   - To flush the surge queue of a virtual server, click Virtual Servers, and then select the virtual server.
   - To flush the surge queue of a service, click Services, and then select the service.

- To flush the surge queue of all the members in a service group, click Service Groups, and then select the service group.
- To flush the surge queue of a specific member in a service group, click Service Groups, and in the action pane, click Manage Members. In the Manage Members of a Service Group dialog box, select the service group member.

Note: You can select multiple entities in any window.

Note: To flush the surge queue of a content switching virtual server, in Steps 1 and 2, expand Content Switching, and then select a virtual server.

3. In the action pane, click Flush Surge Queue.
4. Click OK.

Note: On the appliance, if there are other entities with the same name as you selected, you are alerted that the surge queues of those entities would also be flushed. Take an appropriate action.

Managing a Load Balancing Setup

Jun 09, 2015

An existing Load Balancing setup does not require a great deal of work to maintain as long as it is unchanged, but most do not remain unchanged for long. Increasing load requires new load-balanced servers and eventually new NetScaler appliances, which must be configured and added to the existing setup. Old servers wear out and need to be replaced, requiring removal of some servers and addition of others. Upgrades to your networking equipment or changes to topology may also require modifications to your load balancing setup. Therefore, you will need to perform operations on server objects, services, and virtual servers. The Visualizer can display your configuration graphically, and you can perform operations on the entities in the display. You can also take advantage of a number of other features that facilitate management of the traffic through your load balancing setup.

This section includes the following details:

- Managing Server Objects
- Managing Services
- Managing a Load Balancing Virtual Server
- The Load Balancing Visualizer

Managing Server Objects

Nov 12, 2013

During basic load balancing setup, when you create a service, a server object with the IP address of the service is created, if one does not already exist. If you prefer for your service objects named with domain names rather than IP addresses, you might also have created one or more server objects manually. You can enable, disable, or remove any server object.

When you enable or disable a server object, you enable or disable all services associated with the server object. When you refresh the NetScaler appliance after disabling a server object, the state of its service appears as OUT OF SERVICE. If you specify a wait time when disabling a server object, the server object continues to handle established connections for the specified amount of time, but rejects new connections. If you remove a server object, the service to which it is bound is also deleted.

At the command prompt, type:

enable server <name>@
Example

```
enable server 10.102.29.5
```

1. Navigate to Traffic Management > Load Balancing > Servers.
2. In the details pane, select the server that you want to enable, and then click Enable.
3. In the Enable dialog box, click Yes.

At the command prompt, type:

disable server <name>@ <delay>
Example

```
disable server 10.102.29.5 30
```

1. Navigate to Traffic Management > Load Balancing > Servers.
2. In the details pane, select the server that you want to disable, and then click Disable.
3. In the Wait Time dialog box, type the wait time after which the server is to be disabled (for example 30).
4. Click Enter.

At the command prompt, type:

rm server <name>@
Example

```
rm server 10.102.29.5
```

1. Navigate to Traffic Management > Load Balancing > Servers.
2. In the details pane, select the server that you want to remove, and then click Remove.
3. In the Remove dialog box, click Yes.

Managing Services

Nov 12, 2013

Services are enabled by default when you create them. You can disable or enable each service individually. When disabling a service, you normally specify a wait time during which the service continues to handle established connections, but rejects new ones, before shutting down. If you do not specify a wait time, the service shuts down immediately. During the wait time, the service's state is OUT OF SERVICE.

You can remove a service when it is no longer used. When you remove a service, it is unbound from its virtual server and deleted from the NetScaler configuration.

At the command prompt, type:

enable service <name>

Example

```
enable service Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service that you want to enable (for example, Service-HTTP-1), and click Enable.
3. In the Enable dialog box, click Yes.

At the command prompt, type:

disable service <name>@ <DelayInSeconds>

Example

```
disable service Service-HTTP-1 30
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service that you want to disable (for example, Service-HTTP-1), and then click Disable.
3. In the Wait Time dialog box, type the wait time after which the service is to be disabled (for example, 30).
4. Click Enter.

At the command prompt, type:

rm service <name>@

Example

```
rm service Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service that you want to remove (for example, Service-HTTP-1), and then click Remove.

3. In the Remove dialog box, click Yes.

Managing a Load Balancing Virtual Server

Feb 17, 2014

Virtual servers are enabled by default when you create them. You can disable and enable virtual servers manually. If you disable a virtual server, the virtual server's state appears as OUT OF SERVICE. When this happens, the virtual server terminates all connections, either immediately or after allowing existing connections to complete, depending on the setting of the downStateFLush parameter. If downStateFLush is ENABLED (default), all the connections are flushed. If DISABLED, the virtual server continues to serve requests on existing connections.

You remove a virtual server only when you no longer require the virtual server. Before you remove it, you must unbind all services from it.

At the command prompt, type:

enable lb vserver <name>@

Example

```
enable lb vserver Vserver-LB-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to enable, and click Enable.
3. In the Enable dialog box, click Yes.

At the command prompt, type:

disable vserver

Example

```
disable lb vserver Vserver-LB-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to disable, and then click Disable.
3. In the Disable dialog box, click Yes.

Note: In the disabled state, a virtual server continues to exist on the network. The NetScaler appliance continues to respond to address resolution protocol (ARP) and Internet control message protocol (ICMP) requests directed to the IP address of the virtual server.

At the command prompt, type:

unbind lb vserver <name>@ <serviceName>

Example

```
unbind lb vserver Vserver-LB-1 Service-HTTP-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server from which you want to unbind a service, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, in the Services tab, clear the Active check box next to the service that you want to unbind from the virtual server.
4. Click OK.

At the command prompt, type:

rm lb vserver <name>@

Example

rm lb vserver Vserver-LB-1

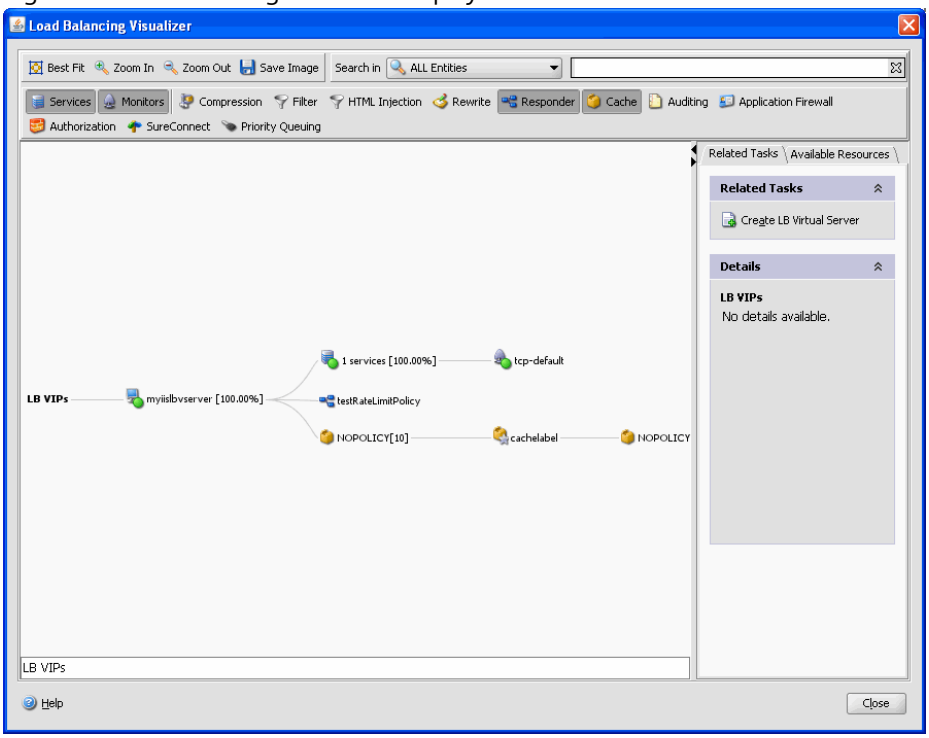
1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to remove, and then click Remove.
3. In the Remove dialog box, click Yes.

The Load Balancing Visualizer

Sep 03, 2013

The Load Balancing Visualizer is a tool that you can use to view and modify the load balancing configuration in graphical format. Following is an example of the Visualizer display

Figure 1. Load Balancing Visualizer Display



You can use the visualizer to view the following:

- The services and service groups that are bound to a virtual server.
- The monitors that are bound to each service.
- The policies that are bound to the virtual server.
- The policy labels, if configured.
- Configuration details of any displayed element.
- Load balancing virtual server statistics.
- Statistical information such as the number of requests received per second by the virtual server and the number of hits per second for rewrite, responder, and cache policies.
- A comparative list of all the parameters whose values either differ or are not defined across service containers.

You can also use the Visualizer to add and bind new objects, modify existing ones, and enable or disable objects. Most configuration elements displayed in the Visualizer appear under the same names as in other parts of the configuration utility. However, unlike the rest of the configuration utility, the Visualizer groups services that have the same configuration details and monitor bindings into an entity called a service container.

A service container is set of similar services and service groups that are bound to a single load balancing virtual server. Next to the service container is a number that shows the number of services in the group. The services in the container have the same properties, with the exception of the name, IP address, and port, and their monitor bindings should have the same weight and binding state. When you bind a new service to a virtual server, it is placed into an existing container if its configuration and monitor bindings match those of other services; otherwise, it is placed in its own container.

The service container display can help you troubleshoot your configuration if something is not functioning as you expect. More than one container for a particular virtual server is an indication that something is wrong with the configuration of that virtual server and its services. To correct the problem, you must first identify the container that has the desired configuration. You can do so by using the Service Attributes Diff feature, described below. After you identify the container, you right-click the container and click Apply Configuration.

The following procedures provide only basic steps for using the Visualizer. Because the Visualizer duplicates functionality in other areas of the Load Balancing feature, other methods of viewing or configuring all of the settings that can be configured in the Visualizer are provided throughout the Load Balancing documentation.

Note: The Visualizer requires a graphic interface, so it is available only through the configuration utility.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to view, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, you can adjust the viewable area as follows:
   - Click the Zoom In and Zoom Out icons to increase or decrease the size of the viewed objects. You can click and drag the viewable area if an item that you want to see disappears from view after zooming in.
   - Click the Best Fit icon to optimize the viewing area.
   - Click the Save Image icon to save the graph as an image file.
   - Click the image, hold down the mouse button, and drag the image to pan the view.
   - In the Search in text field, begin typing the name of the item you are looking for. The item's location is then highlighted. To restrict the search, click the drop-down menu and select the type of element that you want to search for

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to view, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, to view configuration details for entities that are bound to this virtual server, you can do the following:
   - To view a summary of bound services, position the cursor over the virtual server icon.
   - To view services in a service container, click the icon for a service group, click the Related Tasks tab, click Show Member Services, and then click the service group name. To view additional details about the services click Open.
   - To view common properties of services in a service group, click the icon for the service group, click the Related Tasks tab, and view the Details section of the tab.
   - To view a comparative list of the parameters whose values either differ or are not defined across service containers, click the icon for a container, click the Related Tasks tab, and then click Service Attributes Diff. To view monitor binding details for the services in a container, in the Service Attributes Diff dialog box, in the Group column for the container, click Details.
   - To view the details for a monitor, position the cursor over the icon or click the icon for the monitor. For additional details, click the icon, click the Related Tasks tab, and then click View Monitor.
   - To view binding details of a monitor, click the connecting line between the monitor and its related service.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to view, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, to view configuration details for entities that are bound to this virtual server, you can do the following:
   - To view policies that are bound to this virtual server, select one or more policy icons in the tool bar at the top of the dialog box. For example, you can select Compression, Filter, Rewrite, and Responder. If policy labels are configured, they appear in the main view area.
   - For bound policies that appear in the view pane of the Visualizer, to view a policy's expression and actions, position the cursor over the policy icon. To view binding details, position the cursor over the line that connects the policy to the virtual server. To view these details, click the policy. The details of the policy appear in the details pane.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to view, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, to view statistical information, you can do the following:
   - To view detailed statistics for the load balancing virtual server, click the icon for the virtual server, click the Related Tasks tab, and then click Statistics.
   - To view the number of requests received per second at a given point in time by the load balancing virtual server and the number of hits per second at a given point in time for rewrite, responder, and cache policies, click Show Stats. The statistical information is displayed on the respective nodes in the Visualizer. This information is not updated in real time and has to be refreshed manually. To refresh this information, click Refresh Stats.

     Note: The Show Stats option is available only on NetScaler nCore builds.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to view, and then click Visualizer.
3. To copy configuration details for an element to a document or spreadsheet, click the icon for that element, click Related Tasks.
4. In the Related Tasks tab, click Copy Properties and then paste the information into a document.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure bindings, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, click the Available Resources tab, select a resource type in the drop-down menu, and do one or more of the following:
   - To bind a new monitor to a service, select Monitors, click a particular monitor, and then drag it to the service container icon. Use CONTROL + click to select multiple monitors and drag them to the service.
   - To bind a service or service group, select Services or Service Groups, respectively, click a particular service or service group, and then drag it to the virtual server icon. To bind multiple services or service groups at one time, press CONTROL + click to select multiple services and drag them over the virtual server.
   - To bind a policy, select one of the policy groups, click a particular policy, and then drag it to a virtual server. To bind multiple policies (classic policies only) at one time, press CONTROL + policies and drag them over the virtual server. For details on classic and advanced policies, see Policy Configuration and Reference.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server from which you want to unbind a service, policy, or monitor, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, on the Visualizer image, click the connecting line between the resources that you want to unbind, and then click Unbind. For example, to unbind a monitor, you would click the link between the monitor and its bound service and click Unbind.
4. In the Unbind dialog box, click Yes.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to configure, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, on the Visualizer image, double-click the resource that you want to modify.
   Note: Alternatively, on the Available Resources tab, select the resource type from the drop-down menu, select the particular resource that you want to configure and then click Open.
4. In the modify dialog box, enter new settings for the resource.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server that you want to configure, and then click Visualizer.
3. In the Load Balancing Visualizer dialog box, right-click the icon for the resource that you want to add, remove, or disable, and then select the corresponding option from the menu.
   Note: Alternatively, on the Available Resources tab, click the resource type from the drop-down menu, and then click Add to add an entity, or select the particular resource that you want to configure and then click Open.
   Note: These options are not available for service groups or policies.

Managing Client Traffic

Jun 03, 2015

Managing client connections properly helps to ensure that your applications remain available to users even when your NetScaler appliance is experiencing high loads. A number of load balancing features and other features available on the appliance can be integrated into a load balancing setup to process load more efficiently, divert it when necessary, and prioritize the tasks that the appliance must perform:

- **Sessionless load balancing.** You can configure sessionless load balancing virtual servers and perform load balancing without creating sessions in configurations that use DSR or intrusion detection systems (IDS).
- **Integrated caching.** You can redirect HTTP requests to a cache.
- **Priority queuing.** You can direct requests based on priority, by integrating your configuration with the Priority Queuing feature.
- **SureConnect.** You can use load balancing with the SureConnect feature to redirect important requests to a custom Web page, insulating them from delays due to network congestion.
- **Delayed cleanup.** You can configure delayed cleanup of virtual server connections to prevent the cleanup process from using CPU cycles during periods when the NetScaler appliance is experiencing high loads.
- **Rewrite.** You can use the Rewrite feature to modify port and protocol when performing HTTP redirection, or insert the virtual server IP address and port into a custom Request header.
- **RTSP NAT.**
- **Rate-based monitoring.** You can enable rate-based monitoring to divert excess traffic.
- **Layer 2 Parameters.** You can configure a virtual server to use the L2 parameters to identify a connection.
- **ICMP Response**. You can configure the NetScaler to send ICMP responses to PING requests according to your settings. On the IP address corresponding to the virtual server, set the ICMP RESPONSE to VSVR_CNTRLD, and on the virtual server, set the ICMP VSERVER RESPONSE.
  The following settings can be made on a virtual server:
  - When you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.
  - When you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds even if one virtual server is UP.
  - When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds even if one virtual server set to ACTIVE is UP.

To manage client traffic, see the following sections:

- Configuring Sessionless Load Balancing Virtual Servers
- Redirecting HTTP Requests to a Cache
- Directing Requests According to Priority
- Directing Requests to a Custom Web Page
- Enabling Cleanup of Virtual Server Connections
- Graceful Shut down of Services
- Rewriting Ports and Protocols for HTTP Redirection
- Inserting the IP Address and Port of a Virtual Server in the Request Header
- Using a Specified Source IP for Backend Communication
- Setting a Timeout Value for Idle Client Connections
- Managing RTSP Connections
- Managing Client Traffic on the Basis of Traffic Rate
- Identifying a connection with Layer 2 Parameters

- Configuring the Prefer Direct Route Option

Configuring Sessionless Load Balancing Virtual Servers

Nov 11, 2013

When the NetScaler appliance performs load balancing, it creates and maintains sessions between clients and servers. The maintenance of session information places a significant load on the NetScaler resources, and sessions might not be needed in scenarios such as a direct server return (DSR) setup and the load balancing of intrusion detection systems (IDS). To avoid creating sessions when they are not necessary, you can configure a virtual server on the appliance for sessionless load balancing. In sessionless load balancing, the appliance carries out load balancing on a per-packet basis.

Sessionless load balancing can operate in MAC-based forwarding mode or IP-based forwarding mode.

For MAC-based forwarding, the IP address of the sessionless virtual server must be specified on all the physical servers to which the traffic is forwarded.

For IP-based forwarding in sessionless load balancing, the IP address and port of the virtual server need not be specified on the physical servers, because this information is included in the forwarded packets. When forwarding a packet from the client to the physical server, the appliance leaves client details such as IP address and port unchanged and adds the IP address and port of the destination.

NetScaler sessionless load balancing supports the following service types and load balancing methods:

Service Types

- ANY for MAC-based redirection
- ANY, DNS, and UDP for IP-based redirection

Load Balancing Methods

- Round Robin
- Least Bandwidth
- LRTM (Least response time method)
- Source IP Hash
- Destination IP Hash
- Source IP Destination IP Hash
- Source IP Source Port Hash
- Custom Load

Limitations

Sessionless load balancing has the following limitations:

- The NetScaler must be deployed in two-arm mode.
- A service must be bound to only one virtual server.
- Sessionless load balancing is not supported for service groups.
- Sessionless load balancing is not supported for domain based services (DBS services).
- Sessionless load balancing in the IP mode is not supported for a virtual server that is configured as a backup to a primary virtual server.
- You cannot enable spillover mode.

- For all the services bound to a sessionless load balancing virtual server, the Use Source IP (USIP) option must be enabled.
- For a wildcard virtual server or service, the destination IP address will not be changed.

Note: While configuring a virtual server for sessionless load balancing, explicitly specify a supported load balancing method. The default method, Least Connection, cannot be used for sessionless load balancing.
Note: To configure sessionless load balancing in MAC-based redirection mode on a virtual server, the MAC-based forwarding option must be enabled on the NetScaler.

At the command prompt, type the following commands to add a sessionless virtual server and verify the configuration:

- add lb vserver <name>@ <serviceType> <IPAddress>@ <port> -m <redirectionMode> -sessionless <(ENABLED | DISABLED)> -lbMethod <load_balancing_method>
- show lb vserver <name>

Example

```
add lb vserver sesslessv1 any 11.11.12.123 54 -sessionless ENABLED -lbMethod roundrobin -m ip
    Done
show lb vserver sesslessv1
    sesslessv1 (11.11.12.123:54) - ANY Type: ADDRESS
    State: DOWN
    ...
    Effective State: DOWN
    Client Idle Timeout: 120 sec
    Down state flush: ENABLED
    ...
    Persistence: NONE
    Sessionless LB: ENABLED
    Connection Failover: DISABLED
    L2Conn: OFF
    1) Policy : cmp_text Priority:8680 Inherited
    2) Policy : cmp_nocmp_ie60 Priority:8690 Inherited
```

To configure sessionless load balancing on an existing virtual server

At the command prompt, type:

set lb vserver <name>@ -m <redirectionMode> -sessionless <(ENABLED | DISABLED)> -lbMethod <load_balancing_method>

```
set lb vserver sesslessv1 -m mac -sessionless ENABLED -lbmethod lrtm
  Done
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, do one of the following:
   - To add a sessionless virtual server, click Add.
   - To specify sessionless load balancing for an existing virtual server, select it, and then click Open.

3. In the Configure Virtual Server (Load Balancing) dialog box, specify values for the following parameters:
   - Service Name*-serviceName
   - Protocol*-serviceType
   - Server*-serverName
   - Port*-port

   *A required parameter
4. In the Configure Virtual Server (Load Balancing) dialog box, on the Methods and Persistence tab, in the LB Method group, select a supported load balancing method.
5. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, under Redirection Mode, select MAC Based or IP Based.
6. Select Sessionless.
7. Click Create or click OK.
8. In the details pane, click the arrow next to the name of the virtual server and verify the configuration.

Redirecting HTTP Requests to a Cache

Nov 11, 2013

The NetScaler cache redirection feature redirects HTTP requests to a cache. You can significantly reduce the impact of responding to HTTP requests and improve your Web site performance through proper implementation of the cache redirection feature.

A cache stores frequently requested HTTP content. When you configure cache redirection on a virtual server, the NetScaler appliance sends cacheable HTTP requests to the cache, and non-cacheable HTTP requests to the origin Web server.

At the command prompt, type:

set lb vserver <name>@ -cacheable <Value>

Example

```
set lb vserver Vserver-LB-1 -cacheable yes
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure cache redirection, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. Select the Cache Redirection check box, and then click OK.

Directing Requests According to Priority

Nov 11, 2013

The NetScaler appliance supports prioritization of client requests with its priority queuing feature. This feature allows you to designate certain requests, such as those from important clients, as priority requests and sends them to the “front of the line,” so that the appliance responds to them first. This allows you to provide uninterrupted service to those clients through demand surges or DDoS attacks on your Web site.

At the command prompt, type:

set lb vserver <name>@ -pq <Value>

Example

set lb vserver Vserver-LB-1 -pq yes

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure priority queuing, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. Select the PQ check box, and then click OK.

Note: You must configure priority queuing globally for it to function correctly.

Directing Requests to a Custom Web Page

Nov 11, 2013

The NetScaler appliance provides the SureConnect option to ensure that web applications respond despite delays caused by limited server capacity or processing speed. SureConnect does this by displaying an alternative web page of your choice when the server that hosts the primary web page is either unavailable or responding slowly.

To configure SureConnect on a virtual server, you must first configure the alternative content. For information about configuring a SureConnect website, see SureConnect. After you configure the website, enable SureConnect on the load balancing virtual server to put your SureConnect custom web page in use.

Note: For SureConnect to function correctly, you must configure it globally.

At the command prompt, type:

set lb vserver <name>@ -sc <Value>

Example

set lb vserver Vserver-LB-1 -sc yes

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure SureConnect, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. Select the SC check box, and then click OK.

Enabling Cleanup of Virtual Server Connections

Nov 11, 2013

Under certain conditions, you can configure the downStateFlush setting to immediately terminate existing connections when a service or a virtual server is marked DOWN. Terminating existing connections frees resources, and in certain cases speeds recovery of overloaded load balancing setups.

The state of a virtual server depends on the states of the services bound to it. The state of each service depends on the responses of the load balanced servers to probes and health checks sent by the monitors that are bound to that service. Sometimes the load balanced servers do not respond. If a server is slow or busy, monitoring probes can time out. If repeated monitoring probes are not answered within the configured timeout period, the service is marked DOWN.

A virtual server is marked DOWN only when all services bound to it are marked DOWN. When a virtual server goes DOWN, it terminates all connections, either immediately or after allowing existing connections to complete.

You must not enable the downStateFlush setting on those application servers that must complete their transactions. You can enable this setting on Web servers whose connections can safely be terminated when they marked DOWN.

The following table summarizes the effect of this setting on an example configuration consisting of a virtual server, Vserver-LB-1, with two services bound to it, Service-TCP-1 and Service-TCP-2. The virtual server intercepts two connections, C1 and C2, and redirects them to Service-TCP-1 and Service-TCP-2, respectively. In the table, E and D denote the state of the downStateFlush setting: E means Enabled, and D means Disabled.

Vserver-LB-1	Service-TCP-1	State of connections
E	E	Both client and server connections are terminated.
E	D	Both client and server connections are terminated. In case of HTTP services, both client and server connections are terminated only if the transaction is active. If the transaction is not active, only client connections are terminated.
D	E	Both client and server connections are terminated. In case of HTTP services, both client and server connections are terminated only if the transaction is active. If the transaction is not active, only server connections are terminated.
D	D	Neither client nor server connections are terminated.

Note: In case of HTTP services, the downStateFlush setting is effective only when the client is connected to the server. If you want to disable a service only when all the established connections are closed by the server or the client, you can use the graceful shutdown option. For information about the graceful shutdown of a service, see Graceful Shutdown of Services.

To configure the down state flush setting on a virtual server by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -downStateFlush <Value>

Example

set lb vserver Vserver-LB-1 -downStateFlush enabled

To configure the down state flush setting on a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure down state flush, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. Select the Down state flush check box, and then click OK.

Graceful Shut down of Services

Nov 03, 2016

During scheduled network outages such as system upgrades or hardware maintenance, you may have to close or disable some services. To avoid disrupting sessions that have already been established, you can specify a wait time, which places a service in the transition out of service (TROFS) state until the specified wait time expires. The service then enters the out of service (OFS) state.

Often, however, you cannot estimate the amount of time needed for all the connections to a service to complete the existing transactions. If a transaction is unfinished when the wait time expires, shutting down the service may result in data loss. In this case, you can specify graceful shutdown for the service, so that the service is disabled only when all the current active client connections are closed by either the server or the client. See the following table for behavior if you specify a wait time in addition to graceful shutdown.
Persistence is maintained according to the specified method even if you enable graceful shutdown. The system continues to serve all the persistent clients, including new connections from the clients, unless the service is marked DOWN during the graceful shutdown state as a result of the checks made by a monitor.

The following table describes graceful shut down options.

Table 1. Graceful Shut down Options

State	Results
Graceful shutdown is enabled and a wait time is specified.	Service is shut down after the last of the current active client connections is served, even if the wait time has not expired. The appliance checks the status of the connections once every second. If the wait time expires, any open sessions are closed.
Graceful shutdown is disabled and a wait time is specified.	Service is shut down only after the wait time expires, even if all established connections are served before expiration.
Graceful shutdown is enabled and no wait time is specified.	Service is shut down only after the last of the previously established connections is served, regardless of the time taken to serve the last connection.
Graceful shutdown is disabled and no wait time is specified.	No graceful shutdown. Service is shut down immediately after the disable option is chosen or the disable command is issued. (The default wait time is zero seconds.)

To terminate existing connections when a service or a virtual server is marked DOWN, you can use the Down State Flush option. For more information, see Enabling Cleanup of Virtual Server Connections.

To configure graceful shutdown for a service by using the command line interface

At the command prompt, type the following commands to shut down a service gracefully and verify the configuration:

- disable service <name>@ [<delay>] [-graceFul (YES | NO)]
- show service <name>

Example

```
> disable service svc1 6000 -graceFul YES
Done
>show service svc1
svc1 (10.102.80.41:80) - HTTP
State: GOING OUT OF SERVICE (Graceful, Out Of Service in 5998 seconds)
Last state change was at Mon Nov 15 22:44:15 2010
Time since last state change: 0 days, 00:00:01.160
...
Down state flush: ENABLED

1 bound monitor:
1) Monitor Name: tcp-default
State: UP           Weight: 1
Probes: 13898    Failed [Total: 0 Current: 0]
Last response: Probe skipped - live traffic to service.
Response Time: N/A
Done

>show service svc1
svc1 (10.102.80.41:80) - HTTP
State: OUT OF SERVICE
Last state change was at Mon Nov 15 22:44:19 2010
Time since last state change: 0 days, 00:00:03.250
Down state flush: ENABLED

1 bound monitor:
1)  Monitor Name: tcp-default
State: UNKNOWN          Weight: 1
Probes: 13898    Failed [Total: 0 Current: 0]
Last response: Probe skipped - service state OFS.
Response Time: N/A
Done
```

To configure graceful shutdown for a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service, and then click Disable.
3. To delay disabling the service, in the Wait Time dialog box, type the wait time after which the service is to be disabled.
4. To disable the service only after all previously initiated transactions have been completed, check the Graceful Shutdown check box.
5. Click Enter.
6. In the Services pane, you can verify that the service is marked as UP until the wait time expires and after that, it is marked as OUT OF SERVICE.

Rewriting Ports and Protocols for HTTP Redirection

Nov 11, 2013

Virtual servers and the services that are bound to them may use different ports. When a service responds to an HTTP connection with a redirect, you might need to configure the NetScaler appliance to modify the port and the protocol to make sure that the redirection goes through successfully. You do this by enabling and configuring the redirectPortRewrite setting.

This setting affects only HTTP and HTTPS traffic. If this setting is enabled on a virtual server, the virtual server rewrites the port on redirects, replacing the port used by the service with the port used by the virtual server.

If the virtual server or service is of type SSL, you must enable SSL redirect on the virtual server or service. If both the virtual server and service are of type SSL, enable SSL redirect on the virtual server.

The redirectPortRewrite setting can be used in the following scenarios:

- The virtual server is of type HTTP and the services are of type SSL.
- The virtual server is of type SSL and the services are of type HTTP.
- The virtual server is of type HTTP and the services are of type HTTP.
- The virtual server is of type SSL and the services are of type SSL.

Scenario 1: The virtual server is of type HTTP and services are of type SSL. SSL redirect, and optionally port rewrite, is enabled on the service. If port rewrite is enabled, the port of HTTPS URLs is rewritten. HTTP URLs from the server are sent as is to the client.

Redirect URL from the Server	**Redirect URL sent to the Client**
Only SSL redirect is enabled. The virtual server can be configured on any port.	
http://domain.com/	http://domain.com/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	https://domain.com/
https://domain.com:444/	https://domain.com:444/
SSL redirect and port rewrite are enabled. The virtual server is configured on port 80.	
http://domain.com/	http://domain.com/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	https://domain.com/

Redirect URL from the Server https://domain.com:444/	Redirect URL sent to the Client https://domain.com/
SSL redirect and port rewrite are enabled. Virtual server is configured on port 8080.	
http://domain.com/	http://domain.com/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	http://domain.com:8080/
https://domain.com:444/	http://domain.com:8080/

Scenario 2: The virtual server is of type SSL and services are of type HTTP. If port rewrite is enabled, only the port of HTTP URLs is rewritten. HTTPS URLs from the server are sent as is to the client.

Redirect URL from the Server	Redirect URL sent to the Client
SSL redirect is enabled on the virtual server. The virtual server can be configured on any port.	
http://domain.com/	https://domain.com/
http://domain.com:8080/	https://domain.com:8080/
https://domain.com/	https://domain.com/
https://domain.com:444/	https://domain.com:444/
SSL redirect and port rewrite are enabled on the virtual server. The virtual server is configured on port 443.	
http://domain.com/	https://domain.com/
http://domain.com:8080/	https://domain.com/
https://domain.com/	https://domain.com/
https://domain.com:444/	https://domain.com:444/
SSL redirect and port rewrite are enabled. The virtual server is configured on port 444.	

Redirect URL from the Server	Redirect URL sent to the Client
http://domain.com/	https://domain.com:444/
http://domain.com:8080/	https://domain.com:444/
https://domain.com/	https://domain.com/
https://domain.com:445/	https://domain.com:445/

Scenario 3: The virtual server and service are of type HTTP. Port rewrite must be enabled on the virtual server. Only the port of HTTP URLs is rewritten. HTTPS URLs from the server are sent as is to the client.

Redirect URL from the Server	Redirect URL sent to the Client
The virtual server is configured on port 80.	
http://domain.com/	http://domain.com/
http://domain.com:8080/	http://domain.com/
https://domain.com/	https://domain.com/
https://domain.com:444/	https://domain.com:444/
The virtual server is configured on port 8080.	
http://domain.com/	http://domain.com:8080/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	https://domain.com/
https://domain.com:445/	https://domain.com:445/

Scenario 4: The virtual server and service are of type SSL. If port rewrite is enabled, only the port of HTTPS URLs is rewritten. HTTP URLs from the server are sent as is to the client.

Redirect URL from the Server	Redirect URL sent to the Client
SSL redirect is enabled on the virtual server. The virtual server can be configured on any port.	

Redirect URL from the Server http://domain.com/	Redirect URL sent to the Client http://domain.com/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	https://domain.com/
https://domain.com:444/	https://domain.com:444/
SSL redirect and port rewrite are enabled on the virtual server. The virtual server is configured on port 443.	
http://domain.com/	http://domain.com/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	https://domain.com/
https://domain.com:444/	https://domain.com/
SSL redirect and port rewrite are enabled on the virtual server. The virtual server is configured on port 444.	
http://domain.com/	http://domain.com/
http://domain.com:8080/	http://domain.com:8080/
https://domain.com/	https://domain.com:444/
https://domain.com:445/	https://domain.com:444/

To configure HTTP redirection on a virtual server by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -redirectPortRewrite (ENABLED | DISABLED)

Example

set lb vserver Vserver-LB-1 -redirectPortRewrite enabled

To configure HTTP redirection on a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure HTTP redirection, and then click Open.

3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. Select the Redirect Port Rewrite check box, and then click OK.

To configure SSL Redirect on an SSL virtual server or service by using the command line interface

At the command prompt, type:

- set ssl vserver <vServerName> - sslRedirect (ENABLED | DISABLED)
- set ssl service <serviceName> - sslRedirect (ENABLED | DISABLED)

Example

```
set ssl vserver Vserver-SSL-1 -sslRedirect enabled
set ssl service service-SSL-1 -sslRedirect enabled
```

To configure SSL redirection and SSL port rewrite on an SSL virtual server or service by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers or Traffic Management > SSL Offload > Services.
2. In the details pane, select the virtual server or service for which you want to configure SSL redirection, and then click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, click the SSL Settings tab, and then click SSL Parameter.
4. In the Configure SSL Params dialog box, select SSL Redirect. Optionally, select SSL Redirect Port Rewrite.
5. Click OK.
6. In the Configure Virtual Server (SSL Offload) dialog box, click OK.

Inserting the IP Address and Port of a Virtual Server in the Request Header

Nov 11, 2013

If you have multiple virtual servers that communicate with different applications on the same service, you must configure the NetScaler appliance to add the IP address and port number of the appropriate virtual server to the HTTP requests that are sent to that service. This setting allows applications running on the service to identify the virtual server that sent the request.

If the primary virtual server is down and the backup virtual server is up, the configuration settings of the backup virtual server are added to the client requests. If you want the same header tag to be added, regardless of whether the requests are from the primary virtual server or backup virtual server, then you must configure the required header tag on both virtual servers.

Note: This option is not supported for wild card virtual servers or dummy virtual servers.

To insert the IP address and port of the virtual server in the client requests by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -insertVserverIPPort <insertVserverIPPort> [<vipHeader>]

Example

set lb vserver Vserver-LB-1 -insertVserverIPPort VipAddr

To insert the IP address and port of the virtual server in the client requests by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to insert the IP address and port, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. In the Vserver IP Port Insertion list, select the VIPADDR or V6TOV4MAPPING, and then type the port header in a text box next to Vserver IP Port Insertion box.
5. Click OK.

Using a Specified Source IP for Backend Communication

Jun 17, 2014

For communication with the physical servers or other peer devices, the NetScaler appliance uses an IP address owned by it as the source IP address. NetScaler maintains a pool of its IP addresses, and dynamically selects an IP address while connecting with a server. Depending on the subnet in which the physical server is placed, NetScaler decides which IP address to use. This address pool is used for sending traffic as well as monitor probes.

In many situations, you may want the NetScaler to use a specific IP address or any IP address from a specific set of IP addresses for backend communications. The following are a few examples:

- A server can distinguish monitor probes from traffic if the source IP address used for monitor probes belongs to a specific set.
- To improve server security, a server may be configured to respond to requests from a specific set of IP addresses or, sometimes, from a single specific IP address. In such a case, the NetScaler can use only the IP addresses accepted by the server as the source IP address.
- The NetScaler can manage its internal connections efficiently if it can distribute its IP addresses into IP sets and use an address from a set only for connecting to a specific service.

To configure the NetScaler to use a specified source IP address, create net profiles (network profiles) and configure the NetScaler entities to use the profile. A net profile can be bound to load balancing or content switching virtual servers, services, service groups, or monitors. A net profile has NetScaler owned IP addresses (SNIPs) that can be used as the source IP address. It can be a single IP address or a set of IP addresses, referred to as an IP set. If a net profile has an IP set, NetScaler dynamically selects an IP address from the IP set at the time of connection. If a profile has a single IP address, the same IP address is used as the source IP.

If a net profile is bound to a load balancing or content switching virtual server, the profile will be used for sending traffic to all the services bound to it. If a net profile is bound to a service group, NetScaler uses the profile for all the members of the service group. If a net profile is bound to a monitor, NetScaler uses the profile for all the probes sent from the monitor.

Usage of a net profile for sending traffic:

If the Use Source IP Address (USIP) option is enabled, NetScaler uses the IP address of the client and ignores all the net profiles. If the USIP option is not enabled, NetScaler selects the source IP in the following manner:

- If there is no net profile on the virtual server or the service/service group, NetScaler uses the default method.
- If there is a net profile only on the service/service group, NetScaler uses that net profile.
- If there is a net profile only on the virtual server, NetScaler uses the net profile.
- If there is a net profile both on the virtual server and service/service group, NetScaler uses the net profile bound to the service/service group.

Usage of a net profile for sending monitor probes:

For monitor probes, NetScaler selects the source IP in the following manner:

- If there is a net profile bound to the monitor, NetScaler uses the net profile of the monitor. It ignores the net profiles bound to the virtual server or service/service group.
- If there is no net profile bound to the monitor,

- If there is a net profile on the service/service group, NetScaler uses the net profile of the service/service group.
- If there is no net profile even on the service/service group, NetScaler uses the default method of selecting a source IP.

Note: If there is no net profile bound to a service, NetScaler looks for a net profile on the service group if the service is bound to a service group.
To use a specified source IP address for communication, go through the following steps:

1. Create IP sets from the pool of SNIPs owned by the NetScaler. An IP set can consist of both SNIP addresses. For instructions, see Creating IP Sets.
2. Create net profiles. For instructions, see Creating a Net Profile.
3. Bind the net profiles to NetScaler entities. For instructions, see Binding a Net Profile to a NetScaler Entity.

Note: A net profile can have only the IP addresses specified as SNIP on the NetScaler.

Managing Net Profiles

A net profile (or network profile) contains an IP address or an IP set. During communication with physical servers or peers, the NetScaler appliance uses the addresses specified in the profile as the source IP address. For more information on the use of net profiles, see Using a User-specified Source IP Address for Backend Communication.

- For instructions on creating a network profile, see Creating a Network Profile.
- For instructions on binding a network profile to a NetScaler entity, see Binding a Network Profile.

Creating an IP Set

Updated: 2014-06-17

An IP set is a set of IP addresses, which are configured on the NetScaler appliance as Subnet IP addresses (SNIPs) . An IP set is identified with a meaningful name that helps in identifying the usage of the IP addresses contained in it. To create an IP set, add an IP set and bind NetScaler owned IP addresses to it. SNIP addresses can be present in the same IP set. For more information about the use of IP sets, see Using a User-specified Source IP Address for Backend Communication.

To create an IP set by using the command line interface

At the command prompt, type the following commands:

- add ipset <name>
- bind ipset <name> <IPAddress>@
  or
- bind ipset <name> <IPAddress>@
- show ipset [<name>]
  The above command shows the names of all the IP sets on the NetScaler if you do not pass any name. It shows the IP addresses bound to the specified IP set if you pass a name.

Examples

1.

```
> add ipset skpnwipset
 Done
> bind ipset skpnwipset 21.21.20.1
 Done
```

2.

```
 > add ipset testnwipset
 Done
> bind ipset testnwipset 21.21.21.[21-25]
 IPAddress "21.21.21.21" bound
 IPAddress "21.21.21.22" bound
 IPAddress "21.21.21.23" bound
 IPAddress "21.21.21.24" bound
 IPAddress "21.21.21.25" bound
 Done
```

3.

```
 > bind ipset skpipset 11.11.11.101
 ERROR: Invalid IP address
[This IP address could not be added because this is not an IP address owned by the NetScaler]
 > add ns ip 11.11.11.101 255.255.255.0 -type SNIP
 ip "11.11.11.101" added
 Done
 > bind ipset skpipset 11.11.11.101
 IPAddress "11.11.11.101" bound
 Done
```

4.

```
> sh ipset
1) Name: ipset-1
2) Name: ipset-2
3) Name: ipset-3
4) Name: skpnewipset
 Done
```

5.

```
> sh ipset skpnewipset
IP:21.21.21.21
IP:21.21.21.22
IP:21.21.21.23
IP:21.21.21.24
IP:21.21.21.25
 Done
```

To create an IP set by using the configuration utility

1. Navigate to System > Network > IP Sets.
2. In the details pane, do one of the following:
   - To create a new IP set, click Add.
   - To modify an existing IP set, select the IP set, and then click Open.
3. In the Create IP Set dialog box, set the following parameters:
   - Name
   - IP Address (The SNIPs specified on the NetScaler are displayed. Check the IP addresses that you want to bind to the IP set. You can select more than one.)

If you want to add an IP address to the pool, do one of the following:

- To add an IPv4 address, click Add IPv4, and then in the Create IP dialog box, type the necessary details.
- To add an IPv6 address, click Add IPv6, and then in the Create IPV6 dialog box, type the necessary details.

4. Click Create.

Creating a Net Profile

Updated: 2014-06-17

A net profile (network profile) consists of one or more SNIP addresses of the NetScaler. For more information about the usage of net profiles, see Using a User-specified Source IP Address for Backend Communication.

To create a net profile by using the command line interface

At the command prompt, type:

add netprofile <name> [-srcIp <srcIpVal>] If the srcIpVal is not provided in this command, it can be provided later by using the set netprofile command.
Examples

```
> add netprofile skpnetprofile1 -srcIp 21.21.20.1
Done

> add netprofile baksnp -srcIp bakipset
Done

> set netprofile yahnp -srcIp 12.12.23.1
Done

> set netprofile citkbnp -srcIp citkbipset
Done
```

Binding a Net Profile to a NetScaler Entity

Updated: 2013-11-12

A net profile can be bound to a load balancing virtual server, service, service group, or a monitor. For more information about the effect of binding a net profile to a NetScaler entity, see Using a User-specified Source IP Address for Backend Communication.
Note: You can bind a net profile at the time of creating a NetScaler entity or bind it to an already existing entity.

To bind a net profile to a server by using the command line interface

You can bind a net profile to load balancing virtual servers and content switching virtual servers. Specify the appropriate virtual server.

At the command prompt, type:

- set lb vserver <name>@ -netProfile <net_profile_name>
  or
- set cs vserver <name> -netProfile <net_profile_name>

Examples

```
set lb vserver skpnwvs1 -netProfile gntnp
 Done
set cs vserver mmdcsv -netProfile mmdnp
 Done
```

To bind a net profile to a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers or Traffic Management > Content Switching > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind a net profile, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, or Configure Virtual Server (Content Switching) click the Profiles tab.
4. In the Net Profile drop-down list, select a net profile. In this dialog box, you can click New... to add a net profile or Modify... to modify the selected net profile.
5. Click OK.

To bind a net profile to a service by using the command line interface

At the command prompt, type:

set service <name>@ -netProfile <net_profile_name>
Example

```
set service brnssvc1 -netProfile brnsnp
 Done
```

To bind a net profile to a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service to which you want to bind a net profile, and then click Open.
3. In the Configure Service dialog box, click the Profiles tab.
4. In the Net Profile drop-down list, select a net profile. In this dialog box, you can click New... to add a net profile or Modify... to modify the selected net profile.
5. Click OK.

To bind a net profile to a service group by using the command line interface

At the command prompt, type:

set servicegroup <serviceGroupName>@ -netProfile <net_profile_name>
Example

```
set servicegroup ndhsvcgrp -netProfile ndhnp
 Done
```

To bind a net profile to a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group to which you want to bind a net profile, and then click Open.
3. In the Configure Service Group dialog box, click the Profiles tab.
4. In the Net Profile drop-down list, select a net profile. In this dialog box, you can click New... to add a net profile or

Modify... to modify the selected net profile.

5. Click OK.

To bind a net profile to a monitor by using the command line interface

At the command prompt, type:

set monitor <monitor_name> -netProfile <net_profile_name>
Example

```
set monitor brnsecvmon1 -netProfile brnsmonnp
 Done
```

To bind a net profile to a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, select the monitor to which you want to bind a net profile, and then click Open.
3. In the Configure Monitor dialog box, in the Net Profile drop-down list, select a net profile.
4. Click OK.

Setting a Timeout Value for Idle Client Connections

Nov 11, 2013

You can configure a virtual server to terminate any idle client connections after a configured timeout period elapses. When you configure this setting, the NetScaler appliance waits for the time you specify and, if the client is idle after that time, it closes the client connection.

To set a time-out value for idle client connections by using the command line interface

At the command prompt, type:

set lb vserver <name>@ -cltTimeout <Value>

Example

set lb vserver Vserver-LB-1 -cltTimeout 100

To set a time-out value for idle client connections by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure a time-out value, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. In the Client Time-out (secs) text box, type the timeout value (for example, 100).
5. Click OK.

Managing RTSP Connections

Nov 11, 2013

The NetScaler appliance can use either of two topologies—NAT-on mode or NAT-off mode—to load balance RTSP servers. In NAT-on mode, Network Address Translation (NAT) is enabled and configured on the appliance. RTSP requests and responses both pass through the appliance. You must therefore configure the appliance to perform network address translation (NAT) to identify the data connection.

For more information about enabling and configuring NAT, see "IP Addressing."

In NAT-off mode, NAT is not enabled and configured. The appliance receives RTSP requests from the client and routes them to the service that it selects using the configured load balancing method. The load balanced RTSP servers send their responses directly to the client, bypassing the appliance. You must therefore configure the appliance to use Direct Server Return (DSR) mode, and assign publicly accessible FQDNs in DNS to your load balanced RTSP servers.

For more information about enabling and configuring DSR mode, see "Configuring Load Balancing in Direct Server Return Mode." For more information about configuring DNS, see "Domain Name System."

In either case, when you configure RTSP load balancing, you must also configure rtspNat to match the topology of your load balancing setup.

To configure RTSP NAT by using the command line interface

At the command prompt, type:

set lb vserver <name>@–RTSPNAT <ValueOfRTSPNAT>

Example

set lb vserver vserver-LB-1 –RTSPNAT ON

To configure RTSP NAT by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure RTSP NAT, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. Select the RTSP Natting check box, and then click OK.

Managing Client Traffic on the Basis of Traffic Rate

Jun 03, 2015

You can monitor the rate of traffic that flows through load balancing virtual servers and control the behavior of the NetScaler appliance based on the traffic rate. You can throttle the traffic flow if it is too high, cache information based on the traffic rate, and if the traffic rate is too high redirect excess traffic to a different load balancing virtual server. You can apply rate-based monitoring to HTTP and Domain Name System (DNS) requests.

For more information on rate-based policies, see Rate Limiting.

Identifying a connection with Layer 2 Parameters

Nov 12, 2013

Generally, to identify a connection, the NetScaler uses the 4-tuple of client IP address, client port, destination IP address, and destination port. When you enable the L2 Connection option, the Layer 2 parameters of the connection (channel number, MAC address, and VLAN ID) are used in addition to the normal 4-tuple.

Enabling the L2Conn parameter for a load balancing virtual server allows multiple TCP and non-TCP connections with the same 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>) to co-exist on the NetScaler appliance. The appliance uses both the 4-tuple and the Layer 2 parameters to identify TCP and non-TCP connections.

You can enable the L2Conn option in the following scenarios:

- Multiple VLANs are configured on the NetScaler appliance, and a firewall is set up for each VLAN.
- You want the traffic originating from the servers in one VLAN and bound for a virtual server in another VLAN to pass through the firewalls configured for both VLANs.

Therefore, when an nCore NetScaler appliance on which the l2Conn parameter is set for one or more load balancing virtual servers is downgraded to a Classic build or to an nCore build that does not support the l2Conn parameter, the load balancing configurations that use the l2Conn parameter become ineffective.

To configure the L2 connection option by using the command line interface

At the command prompt, type:

add lb vserver <name>@ <serviceType> <IPAddress>@ <port> -l2Conn ON

Example

```
add lb vserver LB-VIP1 HTTP 10.1.1.254 80 -l2Conn ON
```

To configure the L2 connection option by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click **Add**.
3. In the CreateVirtual Server (Load Balancing) dialog box, specify values.: *A required parameter
4. On the Advanced tab, selectL2 Connection.
5. Click Create.
6. Open the virtual server you configured and verify the configuration.

Configuring the Prefer Direct Route Option

Aug 29, 2013

On a wildcard load balancing virtual server if you explicitly configure a route to a destination, by default, the NetScaler appliance forwards traffic according to the configured route. If you want the NetScaler to not look up for the configured route, you can set the Prefer Direct Route option to NO.

If a device is directly connected to a NetScaler appliance, the NetScaler directly forwards traffic to the device. For example, if the destination of a packet is a firewall, the packet need not be routed through another firewall. However, in some cases, you may want the traffic to go through the firewall even if the device is directly connected to it. In such cases, you can set the Prefer Direct Route Option to NO.

Note: The preferDirectRoute setting is applicable to all the wildcard virtual servers on the NetScaler appliance.

To set the prefer direct route option by using the command line interface

At the command prompt, type:

set lb parameter -preferDirectRoute (YES | NO)

Example

set lb parameter -preferDirectRoute YES

To set the prefer direct route option by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. Under Settings, click Configure Load Balancing Parameters.
3. In the Configure Load Balancing Parameters dialog box, select the Prefer Direct Route check box.
4. Click OK.

Advanced Load Balancing Settings

Jun 03, 2015

In addition to configuring virtual servers, you can configure advanced settings for services.

To configure advanced load balancing settings, see the following sections:

- Gradually Stepping Up the Load on a New Service with Virtual Server–Level Slow Start
- The No-Monitor Option for Services
- Protecting Applications on Protected Servers Against Traffic Surges
- Enabling Cleanup of Service Connections
- Directing Requests to a Custom Web Page
- Enabling Access to Services When Down
- Enabling TCP Buffering of Responses
- Enabling Compression
- Maintaining Client Connection for Multiple Client Requests
- Inserting the IP Address of the Client in the Request Header
- Using the Source IP Address of the Client When Connecting to the Server
- Configuring the Source Port for Server-Side Connections
- Setting a Limit on the Number of Client Connections
- Setting a Limit on Number of Requests Per Connection to the Server
- Setting a Threshold Value for the Monitors Bound to a Service
- Setting a Timeout Value for Idle Client Connections
- Setting a Timeout Value for Idle Server Connections
- Setting a Limit on the Bandwidth Usage by Clients
- Redirecting Client Requests to a Cache
- Retaining the VLAN Identifier for VLAN Transparency
- Configuring Automatic State Transition Based on Percentage Health of Bound Services

Gradually Stepping Up the Load on a New Service with Virtual Server–Level Slow Start

Mar 16, 2012

You can configure the NetScaler appliance to gradually increase the load on a service (the number of requests that the service receives per second) immediately after the service is either added to a load balancing configuration or has a state change from DOWN to UP (hereafter, the term "new service" is used for both situations). You can either increase the load manually with load values and intervals of your choice (manual slow start) or configure the appliance to increase the load at a specified interval (automated slow start) until the service is receiving as many requests as the other services in the configuration. During the ramp-up period for the new service, the appliance uses the configured load balancing method.

This functionality is not available globally. It has to be configured for each virtual server. The functionality is available only for virtual servers that use one of the following load balancing methods:

- Round robin
- Least connection
- Least response time
- Least bandwidth
- Least packets
- LRTM (Least Response Time Method)
- Custom load

For this functionality, you need to set the following parameters:

- The new service request rate, which is the amount by which to increase the number or percentage of requests sent to a new service each time the rate is incremented. That is, you specify the size of the increment in terms of either the number of requests per second or the percentage of the load being borne, at the time, by the existing services. If this value is set to 0 (zero), slow start is not performed on new services.
  Note: In automated slow start mode, the final increment is smaller than the specified value if the specified value would place a heavier load on the new service than on the other services.
- The increment interval, in seconds. If this value is set to 0 (zero), the load is not incremented automatically. You have to increment it manually.

With automated slow start, a service is taken out of the slow start phase when one of the following conditions applies:

- The actual request rate is less than the new service request rate.
- The service does not receive traffic for three successive increment intervals.
- The request rate has been incremented 200 times.
- The percentage of traffic that the new service must receive is greater than or equal to 100.

With manual slow start, the service remains in the slow start phase until you take it out of that phase.

Manual Slow Start

If you want to manually increase the load on a new service, do not specify an increment interval for the load balancing virtual server. Specify only the new service request rate and the units. With no interval specified, the appliance does not increment the load periodically. It maintains the load on the new service at the value specified by the combination of the new service request rate and units until you manually modify either parameter. For example, if you set the new service request rate and unit parameters to 25 and "per second," respectively, the appliance maintains the load on the new service at 25 requests per second until you change either parameter. When you want the new service to exit the slow start mode

and receive as many requests as the existing services, set the new service request rate parameter to 0.

As an example, assume that you are using a virtual server to load balance 2 services, Service1 and Service2, in round robin mode. Further assume that the virtual server is receiving 240 requests per second, and that it is distributing the load evenly across the services. When a new service, Service3, is added to the configuration, you might want to increase the load on it manually through values of 10, 20, and 40 requests per second before sending it its full share of the load. The following table shows the values to which you set the three parameters.

Table 1. Parameter Values

Parameter	Value
Interval in seconds	0
New service request rate	10, 20, 40, and 0, at intervals that you choose
Units for the new service request rate	Requests per second

When you set the new service request rate parameter to 0, Service3 is no longer considered a new service, and receives its full share of the load.

Assume that you add another service, Service4, during the ramp-up period for Service3. In this example, Service4 is added when the new service request rate parameter is set to 40. Therefore, Service4 begins receiving 40 requests per second.

The following table shows the load distribution on the services during the period described in this example.

Table 2. Load Distribution on Services when Manually Stepping Up the Load

	new service request rate = 10 req/sec (Service3added)	new service request rate = 20 req/sec	new service request rate = 40 req/sec (Service4added)	new service request rate = 0 req/sec (new services exit slow start mode)
Service1	115	110	80	60
Service2	115	110	80	60
Service3	10	20	40	60
Service4	-	-	40	60
Total req/sec (load	240	240	240	240

on the virtual server)	new service request rate = 10 req/sec	new service request rate = 20 req/sec	new service request rate = 40 req/sec	new service request rate = 0 req/sec
	(Service3added)		(Service4added)	(new services exit slow start mode)

Automated Slow Start

If you want the appliance to increase the load on a new service automatically at specified intervals until the service can be considered capable of handling its full share of the load, set the new service request rate parameter, the units parameter, and the increment interval. When all the parameters are set to values other than 0, the appliance increments the load on a new service by the value of the new service request rate, at the specified interval, until the service is receiving it's full share of the load.

As an example, assume that four services, Service1, Service2, Service3, and Service4, are bound to a load balancing virtual server, vserver1. Further assume that vserver1 receives 100 requests per second, and that it distributes the load evenly across the services (25 requests per second per service). When you add a fifth service, Service5, to the configuration, you might want the appliance to send the new service 4 requests per second for the first 10 seconds, 8 requests per second for the next 10 seconds, and so on, until it is receiving 20 requests per second. For this requirement, the following table shows the values to which you set the three parameters:

Table 3. Parameter Values

Parameter	Value
Interval in seconds	10
Increment value	4
Units for the new service request rate	Requests per second

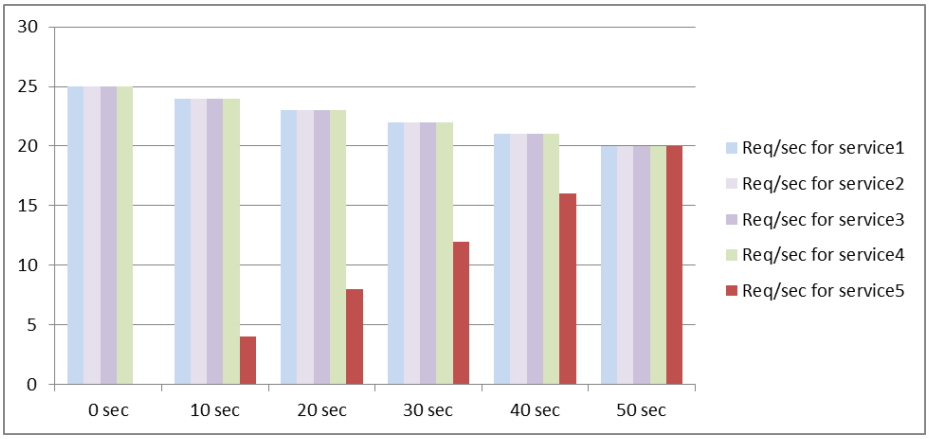
With this configuration, the new service begins receiving as many requests as the existing services 50 seconds after it is added or its state has changed from DOWN to UP. During each interval in this period, the appliance distributes to the existing servers the excess of requests that would have been sent to the new service in the absence of stepwise increments. For example, in the absence of stepwise increments, each service, including Service5, would have received 20 requests each per second. With stepwise increments, during the first 10 seconds, when Service5 receives only 4 requests per second, the appliance distributes the excess of 16 requests per second to the existing services, resulting in the distribution pattern shown in the following table and figure over the 50-second period. After the 50-second period, Service5 is no longer considered a new service, and it receives its normal share of traffic.

Table 4. Load Distribution Pattern on All Services for the 50-second Period Immediately after Service5 is Added

	0 sec	10 sec	20 sec	30 sec	40 sec	50 sec
Req/sec forService1	25	24	23	22	21	20
Req/sec forService2	25	24	23	22	21	20
Req/sec forService3	25	24	23	22	21	20

	0 sec	10 sec	20 sec	30 sec	40 sec	50 sec
Req/sec forService4	25	24	23	22	21	20
Req/sec forService5	0	4	8	12	16	20
Total req/sec (load on the virtual server)	100	100	100	100	100	100

Figure 1. A Graph of the Load Distribution Pattern on All Services for the 50-second Period Immediately after Service5 is Added



An alternative requirement might be for the appliance to send Service5 25% of the load on the existing services in the first 5 seconds, 50% in the next 5 seconds, and so on, until it is receiving 20 requests per second. For this requirement, the following table shows the values to which you set the three parameters.

Table 5. Parameter Values

Parameter	Value
Interval in seconds	5
Increment value	25
Units for the new service request rate	Percent

With this configuration, the service begins receiving as many requests as the existing services 20 seconds after it is added or its state has changed from DOWN to UP. The traffic distribution during the ramp-up period for the new service is identical to the one described earlier, where the unit for the step increments was “requests per second.”

Setting the Slow Start Parameters

Updated: 2013-12-04

You set the slow start parameters by using either the set lb vserver or the add lb vserver command. The following command is for setting slow start parameters when adding a virtual server.

To configure stepwise load increments for a new service by using the command line interface

At the command prompt, type the following commands to configure stepwise increments in the load for a service and verify the configuration:

- add lb vserver <name> <serviceType> <IPAddress> <port> [-newServiceRequest <positive_integer>] [<newServiceRequestUnit>] [-newServiceRequestIncrementInterval <positive_integer>]
- show lb vserver <name>

Example

```
> set lb vserver BR_LB -newServiceRequest 5 PER_SECOND -newServiceRequestIncrementInterval 10
 Done
> show lb vserver BR_LB
 BR_LB (192.0.2.33:80) - HTTP Type: ADDRESS
 State: UP
            ...
            ...
 New Service Startup Request Rate: 5 PER_SECOND, Increment Interval: 10
            ...
            ...
Done
>
```

To configure stepwise load increments for a new service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, do one of the following:
   - To configure stepwise increments for a new load balancing virtual server, click Add.
   - To configure stepwise increments for an existing load balancing virtual server, select the virtual server, and then click Open.
3. In the Create Virtual Server or Configure Virtual Server dialog box, on the Method and Persistence tab, set the following parameters:
   - New Service Startup Request Rate. Also, from the list next to the text box, select the unit (PER_SECOND or PERCENT).
   - Increment Interval.
4. Click Create or OK, and then click Close.

The No-Monitor Option for Services

Nov 11, 2013

If you use an external system to perform health checks on the services and do not want the NetScaler appliance to monitor the health of a service, you can set the no-monitor option for the service. If you do so, the appliance does not send probes to check the health of the service but shows the service as UP. Even if the service goes DOWN, the appliance continues to send traffic from the client to the service as specified by the load balancing method.

The monitor can be in the ENABLED or DISABLED state when you set the no-monitor option, and when you remove the no-monitor option, the earlier state of the monitor is resumed.

You can set the no-monitor option for a service when creating the service. You can also set the no-monitor option on an existing service.

The following are the consequences of setting the no-monitor option:

- If a service for which you enabled the no-monitor option goes down, the NetScaler continues to show the service as UP and continues to forward traffic to the service. A persistent connection to the service can worsen the situation. In that case, or if many services shown as UP are actually DOWN, the system may fail. To avoid such a situation, when the external mechanism that monitors the services reports that a service that is DOWN, remove the service from the NetScaler configuration.
- If you configure the no-monitor option on a service, you cannot configure load balancing in the Direct Server Return (DSR) mode. For an existing service, if you set the no-monitor option, you cannot configure the DSR mode for the service.

To set the no-monitor option for a new service by using the command line interface

At the command prompt, type the following commands to create a service with the health monitor option, and verify the configuration:

add service <serviceName> <IP | serverName> <serviceType> <port> -healthMonitor (YES|NO)

Example

```
 >add service nomonsrvc 10.102.21.21 http 80
 -healthMonitor no
 Done
 > show service nomonsrvc
nomonsrvc (10.102.21.21:80) - HTTP
State: UP
Last state change was at Mon Nov 15 22:41:29 2010
Time since last state change: 0 days, 00:00:00.970
Server Name: 10.102.21.21
 Server ID : 0 Monitor Threshold : 0
...
Access Down Service: NO
...
Down state flush: ENABLED
Health monitoring: OFF
```

1 bound monitor:
1) Monitor Name: tcp-default
State: UNKNOWN Weight: 1
Probes: 3 Failed [Total: 3 Current: 3]
Last response: Probe skipped - Health monitoring is turned off.
Response Time: N/A
Done

To set the no-monitor option for an existing service by using the command line interface

At the command prompt, type the following command to set the health monitor option:

set service <name> -healthMonitor (YES | NO)

Example

By default, the state of a service and the state of the corresponding monitor are UP.
>show service LB-SVC1
LB-SVC1 (10.102.29.5:80) - HTTP
State: UP

1) Monitor Name: http-ecv
State: UP Weight: 1
Probes: 99992 Failed [Total: 0 Current: 0]
Last response: Success - Pattern found in response.
Response Time: 3.76 millisec
Done

When the no-monitor option is set on a service, the state of the monitor changes to UNKNOWN.
> set service LB-SVC1 -healthMonitor NO
Done
> show service LB-SVC1
LB-SVC1 (10.102.29.5:80) - HTTP
State: UP
Last state change was at Fri Dec 10 10:17:37 2010.
Time since last state change: 5 days, 18:55:48.710
Health monitoring: OFF

1) Monitor Name: http-ecv
State: UNKNOWN Weight: 1
Probes: 100028 Failed [Total: 0 Current: 0]
Last response: Probe skipped - Health monitoring is turned off.
Response Time: 0.0 millisec
Done
When the no-monitor option is removed, the earlier state of the monitor is resumed.
> set service LB-SVC1 -healthMonitor YES
Done
>show service LB-SVC1
LB-SVC1 (10.102.29.5:80) - HTTP

State: UP
Last state change was at Fri Dec 10 10:17:37 2010
Time since last state change: 5 days, 18:57:47.880
1) Monitor Name: http-ecv
State: UP Weight: 1
Probes: 100029 Failed [Total: 0 Current: 0]
Last response: Success - Pattern found in response.
Response Time: 5.690 millisec
Done

To set the no-monitor option for a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, do one of the following:
   - To create a new service, click Add.
   - To modify an existing service, select the service and then click Open.
3. In the Create Service or Configure Service dialog box, specify values for the following parameters:
   - Service Name*–serviceName
   - Protocol*–serviceType
   - Server*–ipAddress
   - Port*–port
   - Health Monitor–healthMonitor

   * A required parameter
4. Click Create. The service you created appears in the Services pane.
5. From the Services pane, open the service that you added, and verify the health monitor setting.

Protecting Applications on Protected Servers Against Traffic Surges

Jun 08, 2015

The NetScaler provides the surge protection option to maintain the capacity of a server or cache. The NetScaler regulates the flow of client requests to servers and controls the number of clients that can simultaneously access the servers. The NetScaler blocks any surges passed to the server, thereby preventing overloading of the server.

For surge protection to function correctly, you must enable it globally. For more information about surge protection, see "Surge Protection."

To set surge protection on the service by using the command line interface

At the command prompt, type:

set service <name>@ -sp <Value>

Example

set service Service-HTTP-1 -sp ON

To set surge protection on the service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure surge protection, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab, scroll down, and under Others, select the Surge Protection check box.
4. Click OK.

Enabling Cleanup of Service Connections

Nov 25, 2013

When cleanup of service connections is enabled, the NetScaler performs a cleanup of the connections on a service that is down. This setting is described in Enabling Cleanup of Virtual Server Connections.

To set down state flush on the service by using the command line interface

At the command prompt, type:

set service <name> -downStateFlush <Value>

Example

```
set service Service-HTTP-1 -downStateFlush enabled
```

To set down state flush on the service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure down state flush, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Scroll down, and under Others, select the Down state flush check box.
5. Click OK.

Directing Requests to a Custom Web Page

Jun 08, 2015

For SureConnect to function correctly, you must set it globally. The NetScaler provides the SureConnect option to ensure the response from an application. For more information about the SureConnect option, see "Sure Connect."

To set SureConnect on the service by using the command line interface

At the command prompt, type:

set service <name>@ -sc <Value>

Example

set service Service-HTTP-1 -sc ON

To set SureConnect on the service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure SureConnect, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Scroll down, and under Others, select the Sure Connect check box.
5. Click OK.

Enabling Access to Services When Down

Jun 08, 2015

You can enable access to a service when it is disabled or in a DOWN state by configuring the NetScaler appliance to use Layer 2 mode to bridge the packets sent to the service. Normally, when requests are forwarded to services that are DOWN, the request packets are dropped. When you enable the Access Down setting, however, these request packets are sent directly to the load balanced servers.

For more information about Layer 2 and Layer 3 modes, see IP Addressing.

For the appliance to bridge packets sent to the DOWN services, enable Layer 2 mode with the accessDown parameter.

To enable access down on a service by using the command line interface

At the command prompt, type:

set service <name>@ -accessDown <Value>

Example

set service Service-HTTP-1 -accessDown YES

To enable access down on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure access down, click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Scroll down, and under Others, select the Access Down check box.
5. Click OK.

Enabling TCP Buffering of Responses

Jun 08, 2015

The NetScaler appliance provides a TCP buffering option that buffers only responses from the load balanced server. This enables the appliance to deliver server responses to the client at the maximum speed that the client can accept them. The appliance allocates from 0 through 4095 megabytes (MB) of memory for TCP buffering, and from 4 through 20480 kilobytes (KB) of memory per connection.

Note: TCP buffering set at the service level takes precedence over the global setting. For more information about configuring TCP buffering globally, see "TCP Buffering."

To enable TCP Buffering on a service by using the command line interface

At the command prompt, type:

set service <name>@ -TCPB <Value>

Example

set service Service-HTTP-1 -TCPB YES

To enable TCP Buffering on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure TCP buffering (for example, Service-HTTP-1), and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Scroll down, and under Settings, select the TCP Buffering check box.
5. Click OK.

Enabling Compression

Jun 08, 2015

The NetScaler appliance provides a compression option to transparently compress HTML and text files by using a set of built-in compression policies. Compression reduces bandwidth requirements and can significantly improve server responsiveness in bandwidth-constrained setups. The compression policies are associated with services bound to the virtual server. The policies determine whether a response can be compressed and send compressible content to the appliance, which compresses it and sends it to the client.

Note: For compression to function correctly, you must enable it globally. For more information about configuring compression globally, see Compression.

To enable compression on a service by using the command line interface

At the command prompt, type:

set service <name> -CMP <YES | NO>

Example

set service Service-HTTP-1 -CMP YES

To enable compression on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure compression (for example, Service-HTTP-1), and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Settings, select the Compression check box.
5. Click OK.

Maintaining Client Connection for Multiple Client Requests

Jun 08, 2015

You can set the client keep-alive parameter to configure an HTTP or SSL service to keep a client connection to a Web site open across multiple client requests. If client keep-alive is enabled, even when the load balanced Web server closes a connection, the NetScaler appliance keeps the connection between the client and itself open. This setting allows services to serve multiple client requests on a single client connection.

If you do not enable this setting, the client will open a new connection for every request that it sends to the Web site. The client keep-alive setting saves the packet round trip time required to establish and close connections. This setting also reduces the time to complete each transaction. Client keep-alive can be enabled only on HTTP or SSL service types.

Client keep-alive set at the service level takes precedence over the global client keep-alive setting. For more information about client keep-alive, see Client Keep-Alive.

To enable client keep-alive on a service by using the command line interface

At the command prompt, type:

set service <name>@ -CKA <Value>

Example

set service Service-HTTP-1 -CKA YES

To enable client keep-alive on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure client keep-alive, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Settings, select the Client Keep-Alive check box.
5. Click OK.

Inserting the IP Address of the Client in the Request Header

Nov 11, 2013

A NetScaler uses the mapped IP address (MIP) to connect to the server. The server need not be aware of the client.

However, in some situations, the server needs to be aware of the client it has to serve. When you enable the client IP setting, the NetScaler inserts the client's IPv4 or IPv6 address while forwarding the requests to the server. The server inserts this client IP in the header of the responses. The server is thus aware of the client.

At the command prompt, type:

set service <name>@ -CIP <Value> <cipHeader>

Example

```
set service Service-HTTP-1 -CIP enabled X-forwarded-for
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to add the client IP address in the request, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Settings, select the Client IP check box.
5. In the Header text box, type the header tag (for example, X-Forwarded-for).
6. Click OK.

Using the Source IP Address of the Client When Connecting to the Server

Jun 08, 2015

You can configure the NetScaler appliance to forward packets from the client to the server without changing the source IP address. This is useful when you cannot insert the client IP address into a header, such as when working with non-HTTP services.

For more information about configuring USIP globally, see "Enabling Use Source IP Mode."

For information about using the port of the client when connecting to the server, see Using the Client Port When Connecting to the Server.

At the command prompt, type:

set service <name>@ -usip (**YES** | **NO**)

Example

```
set service Service-HTTP-1 -usip YES
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to enable the USIP mode, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Settings, select the Use Source IP check box.
5. Click OK.

Configuring the Source Port for Server-Side Connections

Jun 08, 2015

When the NetScaler appliance connects to a physical server, it can use the source port from client's request, or it can use a proxy port as the source port for the connection. You can set the Use Proxy Port parameter to YES to handle situations such as the following scenario:

- The NetScaler appliance is configured with two load balancing virtual servers, LBVS1 and LBVS2.
- Both the virtual servers are bound to the same service, S-ANY.
- Use (the client's) source IP address (USIP) is enabled on the service.
- Client C1 sends two requests, Req1 and Req2, for the same service.
- Req1 is received by LBVS1 and Req2 is received by LBVS2.
- LBVS1 and LBVS2 forward the request to S-ANY, and when S-ANY sends the response, they forward the response to the client.
- Consider two cases:
  - Use the client port. When the NetScaler uses the client port, both the virtual servers use the client's IP address (because USIP is ON) and the client's port when connecting to the server. Therefore, when the service sends the response, the NetScaler cannot determine which virtual server should receive the response.
  - Use proxy port. When the NetScaler uses a proxy port, the virtual servers use the client's IP address (because USIP is ON), but different ports when connecting to the server. Therefore, when the service sends the response, the port number identifies the virtual server that should receive the response.

However, if you require a fully transparent configuration, such as a fully transparent cache redirection configuration, you must disable the Use Proxy port Setting so that the NetScaler appliance can use the source port from the client's request.

The Use Proxy Port option becomes relevant if the use source IP (USIP) option is enabled. For TCP-based service types, such as TCP, HTTP, and SSL, the option is enabled by default. For UDP-based service types, such as UDP and DNS, including ANY, the option is disabled by default. For more information about the USIP option, see "Enabling Use Source IP Mode."

You can configure the Use Proxy Port setting either globally or on a given service.

Updated: 2013-11-11

You configure the Use ProxyPort settingon the service if you want to override the global setting.

To configure the Use Proxy Port setting on a service by using the command line interface

At the command prompt, type:

set service <name>@ -useProxyPort (YES | NO)

Example

```
> set service svc1 -useproxyport YES
 Done > show service svc1
```

```
svc1 (10.102.29.30:80) - HTTP
State: UP
. . .
Use Source IP: YES  Use Proxy Port: YES
. . .
Done
>
```

To configure the Use Proxy Port setting on a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to use the source IP address, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Others, in the Use Proxy Port drop-down list, select YES.
5. Click OK.

Updated: 2013-09-13

You configure the Use Proxy Port setting globally if you want to apply the setting to all the services on the NetScaler appliance. The global setting is overridden by service-specific Use Proxy Port settings.

To configure the Use Proxy Port setting globally by using the command line interface

At the command prompt, type the following commands to configure the Use Proxy Port setting globally and verify the configuration:

- set ns param -useproxyport (ENABLED | DISABLED)
- show ns param

```
> set ns param -useproxyport ENABLED
Done
> show ns param
Global configuration settings:
          . . .
          Use Proxy Port: ENABLED
Done
>
```

To configure the Use Proxy Port setting globally by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Settings, click Change global system settings.
3. In the Set the Global Settings Parameters dialog box, select or clear the Use Proxy Port check box.
4. Click OK.

Setting a Limit on the Number of Client Connections

Nov 11, 2013

You can specify a maximum number of client connections that each load balanced server can handle. The NetScaler appliance then opens client connections to a server only until this limit is reached. When the load balanced server reaches its limit, monitor probes are skipped, and the server is not used for load balancing until it has finished processing existing connections and frees up capacity.

For more information on the Maximum Client setting, see "Load Balancing Domain-Name Based Services."

Note: Connections that are in the process of closing are not considered for this limit.

At the command prompt, type:

set service <name> -maxclient <Value>

Example

```
set service Service-HTTP-1 -maxClient 1000
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure the maximum number of client connections (for example, Service-HTTP-1), and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Thresholds, in the Max Clients text box, type the maximum number of client connections (for example, 100).
5. Click OK.

Setting a Limit on Number of Requests Per Connection to the Server

Aug 29, 2013

The NetScaler appliance can be configured to reuse connections to improve performance. In some scenarios, however, load balanced Web servers may have issues when connections are reused for too many requests. For HTTP or SSL services, use the max request option to limit the number of requests sent through a single connection to a load balanced Web server.

Note: You can configure the max request option for HTTP or SSL services only.

At the command prompt, type:

set service <ServiceName> -maxReq <Value>

Example

```
set service Service-HTTP-1 -maxReq 100
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure the maximum number of client requests, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Thresholds, in the Max Requests text box, type the maximum number of client requests (for example, 100).
5. Click OK.

Setting a Threshold Value for the Monitors Bound to a Service

Nov 11, 2013

The NetScaler appliance designates a service as UP only when the sum of the weights of all monitors bound to it and that are UP is equal to or greater than the threshold value configured on the service. The weight for a monitor specifies how much that monitor contributes to designating the service to which it is bound as UP.

For example, assume that three monitors, named Monitor-HTTP-1, Monitor-HTTP-2, and Monitor-HTTP-3 respectively, are bound to Service-HTTP-1, and that the threshold configured on the service is three. Suppose the following weights are assigned to each monitor:

- The weight of Monitor-HTTP-1 is 1.
- The weight of Monitor-HTTP-2 is 3.
- The weight of Monitor-HTTP-3 is 1.

The service is marked UP only when one of the following is true:

- Monitor-HTTP-2 is UP.
- Monitor-HTTP-2 and Monitor-HTTP-1 or Monitor-HTTP-3 are UP
- All three monitors are UP.

At the command prompt, type:

set service <name> -monThreshold <Value>

Example

```
set service Service-HTTP-1 -monThreshold 100
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure monitor threshold (for example, Service-HTTP-1), and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. In the Monitor Threshold text box, type the monitor threshold.
5. Click OK.

Setting a Timeout Value for Idle Client Connections

Nov 11, 2013

You can configure the service with a time-out value to terminate any idle client connections when the configured time elapses. If the client is idle during the configured time, the NetScaler closes the client connection.

At the command prompt, type:

set service <name> -cltTimeout <Value>

Example

```
set service Service-HTTP-1 -cltTimeout 100
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure the time-out value for client connections, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Idle Time-out (secs), in the Client text box, type the timeout value.
5. Click OK.

Setting a Timeout Value for Idle Server Connections

Nov 11, 2013

You can configure a service with a timeout value to terminate any idle server connections when the configured time elapses. If the server is idle for the configured amount of time, the NetScaler appliance closes the server connection.

At the command prompt, type:

set service <name>@ -svrTimeout <Value>

Example

set service Service-HTTP-1 -svrTimeout 100

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure the timeout value for server connections (for example, Service-HTTP-1), and click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Idle Time-out (secs), in the Server text box, type a timeout value as a number of seconds (for example, 100).
5. Click OK.

Setting a Limit on the Bandwidth Usage by Clients

Nov 11, 2013

In some cases, servers may have limited bandwidth to handle client requests and may become overloaded. To prevent overloading a server, you can specify a maximum limit on the bandwidth processed by the server. The NetScaler appliance forwards requests to a load balanced server only until this limit is reached.

At the command prompt, type:

set service <name> -maxBandwidth <Value>

Example

```
set service Service-HTTP-1 -maxBandwidth 100
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details page, select the service for which you want to configure maximum bandwidth usage (for example, Service-HTTP-1), and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Under Thresholds, in the Max Bandwidth (kbits) text box, type the maximum bandwidth (for example, 100).
5. Click OK.

Redirecting Client Requests to a Cache

Nov 11, 2013

You can configure a service to redirect client requests to a cache, and forward only those requests that are cache misses to a service chosen by the configured load balancing method.

At the command prompt, type:

set service <name>@ -cacheable <Value>

Example

```
set service Service-HTTP-1 -cacheable YES
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to configure cache redirection, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab.
4. Scroll down, and under Cache Redirection Options, in Cache Type list, select the type of cache (for example, Regular Server).
5. Click OK.

Retaining the VLAN Identifier for VLAN Transparency

Aug 29, 2013

You can configure a load balancing virtual server to retain the client's VLAN identifier in packets that are to be forwarded to servers. The virtual server must be a wildcard virtual server of type ANY, and must be functioning in MAC mode.

At the command prompt, type the following command to configure a load balancing virtual server to retain the client VLAN ID and verify the configuration:

- set lb vserver <name> -m MAC -macmodeRetainvlan ENABLED
- show lb vserver <name>

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. Select the virtual server for which you want to configure VLAN transparency, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, select the Macmode Retain VLAN check box.
4. Click OK.

Configuring Automatic State Transition Based on Percentage Health of Bound Services

Nov 11, 2013

You can configure a load balancing virtual server to automatically transition from the UP state to the DOWN state if the percentage of active services falls below a configured threshold. For example, if you bind 10 services to a load balancing virtual server and configure a threshold of 50% for that virtual server, it transitions from UP to DOWN if six or more services are DOWN. When the percentage health rises above the threshold value, the virtual server returns to the UP state.

You can also enable an SNMP alarm called ENTITY-STATE if you want the NetScaler appliance to notify you when the percentage health of bound services causes a virtual server to change state.

At the command prompt, type the following commands to configure automatic state transition for a virtual server and verify the configuration:

- set lb vserver <name>@ -healthThreshold <positive_integer>
- show lb vserver <name>

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. Select the load balancing virtual server that you want to configure, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. In the Health Threshold text box, type the threshold value. The threshold value is an integer ranging from 0 (the default) to 100.
5. Click OK.

At the command prompt, type the following commands to enable the ENTITY-STATE SNMP alarm and verify the configuration:

- enable snmp alarm ENTITY-STATE
- show snmp alarm

1. In the navigation pane, expand System, expand SNMP, and then click Alarms.
2. In the details pane, click the ENTITY-STATE alarm, and then, in the Action list, click Enable.

The Built-in Monitors

Jun 09, 2015

The NetScaler appliance contains a number of built-in monitors that you can use to monitor your services. These built-in monitors handle most of the common protocols. You cannot modify or remove the built-in monitors; you can only bind a built-in monitor to a service and unbind it from the service.

Note: You can create a custom monitor based on a built-in monitor. To learn how to create custom monitors, see Configuring Monitors in a Load Balancing Setup.

This section includes the following details:

- Monitoring TCP-based Applications
- Monitoring SSL Services
- Monitoring FTP Services
- Monitoring SIP Services
- Monitoring RADIUS Services
- Monitoring Accounting Information Delivery from a RADIUS Server
- Monitoring DNS and DNS-TCP Services
- Monitoring LDAP Services
- Monitoring MySQL Services
- Monitoring SNMP Services
- Monitoring NNTP Services
- Monitoring POP3 Services
- Monitoring SMTP Services
- Monitoring RTSP Servers
- Monitoring the XML Broker Services
- Monitoring ARP Requests
- Monitoring the XenDesktop Delivery Controller Services
- Monitoring Web Interface Services
- Monitoring Citrix StoreFront Stores

Monitoring TCP-based Applications

Apr 03, 2014

The NetScaler appliance has two built-in monitors that monitor TCP-based applications: tcp-default and ping-default. When you create a service, the appropriate default monitor is bound to it automatically, so that the service can be used immediately if it is UP. The tcp-default monitor is bound to all TCP services; the ping-default monitor is bound to all non-TCP services.

You cannot delete or modify default monitors. When you bind any other monitor to a TCP service, the default monitor is unbound from the service. The following table lists the monitor types, and the parameters and monitoring processes associated with each type.

Monitor type	Specific parameters	Process
tcp	Not applicable	The NetScaler appliance establishes a 3-way handshake with the monitor destination, and then closes the connection. If the appliance observes TCP traffic to the destination, it does not send TCP monitoring requests. This occurs if LRTM is disabled. By default, LRTM is disabled on this monitor.
http	httprequest [“HEAD /”] - HTTP request that is sent to the service. respcode [200] - A set of HTTP response codes are expected from the service.	The NetScaler appliance establishes a 3-way handshake with the monitor destination. After the connection is established, the appliance sends HTTP requests, and then compares the response code with the configured set of response codes.
tcp-ecv	send [""] - is the data that is sent to the service. The maximum permissible length of the string is 512 K bytes. recv [""] - expected response from the service. The maximum permissible length of the string is 128 K bytes.	The NetScaler appliance establishes a 3-way handshake with the monitor destination. When the connection is established, the appliance uses the send parameter to send specific data to the service and expects a specific response through the receive parameter.

Monitor type	Specific parameters	Process
http-ecv	send [""] - HTTP data that is sent to the service recv [""] - the expected HTTP response data from the service	The NetScaler appliance establishes a 3-way handshake with the monitor destination. When the connection is established, the appliance uses the send parameter to send the HTTP data to the service and expects the HTTP response that the receive parameter specifies. (HTTP body part without including HTTP headers). Empty response data matches any response. Expected data may be anywhere in the first 24K bytes of the HTTP body of the response.
ping	Not Applicable	The NetScaler appliance sends an ICMP echo request to the destination of the monitor and expects an ICMP echo response.

To configure built-in monitors for TCP-based applications, see Configuring Monitors in a Load Balancing Setup.

Monitoring SSL Services

Jun 16, 2014

The NetScaler appliance has built-in secure monitors, TCPS and HTTPS. You can use the secure monitors to monitor HTTP as well as non-HTTP traffic. To configure a secure HTTP monitor, select the monitor type as HTTP, and then set the secure flag. To configure a secure TCP monitor, select the monitor type as TCP, and then set the secure flag.The secure monitors work as described below:

- **Secure TCP monitoring.** The NetScaler appliance establishes a TCP connection. After the connection is established, the appliance performs an SSL handshake with the server. After the handshake is over, the appliance closes the connection.
- **Secure HTTP monitoring.** The NetScaler appliance establishes a TCP connection. After the connection is established, the appliance performs an SSL handshake with the server. When the SSL connection is established, the appliance sends HTTP requests over the encrypted channel and checks the response codes.

The following table describes the available built-in monitors for monitoring SSL services.

Monitor type	Probe	Success criteria (Direct condition)
TCP	TCP connection SSL handshake	Successful TCP connection established and successful SSL handshake.
HTTP	TCP connection SSL handshake Encrypted HTTP request	Successful TCP connection is established, successful SSL handshake is performed, and expected HTTP response code in server HTTP response is encrypted.
TCP-ECV	TCP connection SSL handshake (Data sent to a server is encrypted.)	Successful TCP connection is established, successful SSL handshake is performed, and expected TCP data is received from the server.
HTTP-ECV	TCP connection SSL handshake (Encrypted HTTP request)	Successful TCP connection is established, successful SSL handshake is performed, and expected HTTP data is received from the server.

Monitoring FTP Services

Mar 24, 2015

To monitor FTP services, the NetScaler appliance opens two connections to the FTP server. It first connects to the control port, which is used to transfer commands between a client and an FTP server. After it receives the expected response, it connects to the data port, which is used to transfer files between a client and an FTP server. Only when the FTP server responds as expected on both connections is it marked UP.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.
The NetScaler appliance has two built-in monitors for FTP services: the FTP monitor and the FTP-EXTENDED monitor. The FTP monitor checks basic functionality; the FTP-EXTENDED monitor also verifies that the FTP server is able to transmit a file correctly.

Parameter	Specifies
userName	User name used in the probe. Applies to both the FTP and FTP-EXTENDED monitor.
password	Password used in monitoring. Applies to both the FTP and FTP-EXTENDED monitor
fileName	File name to be used for FTP-EXTENDED monitor only.

To configure built-in monitors to check the state of FTP services, see Configuring Monitors in a Load Balancing Setup.

Monitoring SIP Services

Oct 14, 2014

A NetScaler ADC has two built-in monitors that you can use to monitor SIP services: the **SIP-UDP** and **SIP-TCP** monitors. A SIP monitor periodically checks the SIP service to which the SIP monitor is bound, by sending SIP request methods to the SIP service. If the SIP service replies with a response code, the monitor marks the service as UP. If the SIP service does not respond, or responds incorrectly, it is marked as DOWN.

Parameter	Specifies
sipURI	SIP addressing schema of the SIP server.
sipmethod	Type of SIP request used to probe the SIP service. Specify one of the following methods: • INVITE • OPTION (the default) • REGISTER
respcode	SIP response code with which the SIP service responds the probe request. Default: 200.

The Session Initiation Protocol (SIP) is designed to initiate, manage, and terminate multimedia communications sessions. It has emerged as the standard for Internet telephony (VoIP). SIP messages can be transmitted over TCP or UDP. SIP messages are of two types: request messages and response messages.

The following table summarizes the structure of SIP messages.

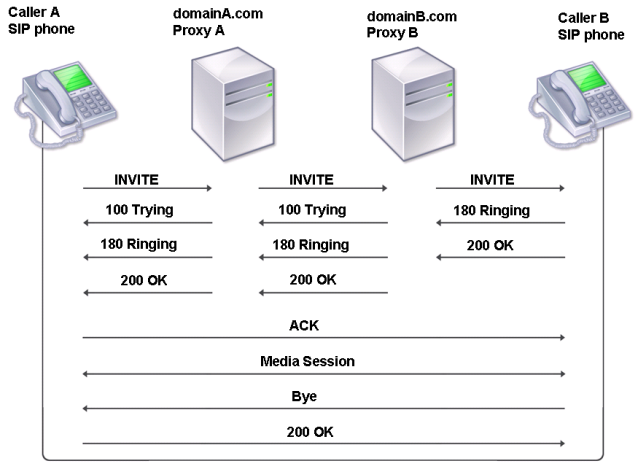
Message type	Components	Components
Request	Method	Invite, Ack, Options, Bye, Cancel, Register
	Request URI	Represents the subject, media type, or urgency of sessions initiated. The common format is: sip:user:password@host:port;uri-parameters?headers
	SIP version	The SIP version being used
Response	SIP version	The SIP version that is being used.
	Status code	A 3-digit integer result code. The possible values are: 1xx: Information Responses. For example: 180, Ringing

Message type	Components	Components
		2xx: Successful Responses. For example: 200, OK 3xx: Redirection Responses. For example: 302, Moved Temporarily 4xx: Request Failures Responses. For example: 403, Forbidden 5xx: Server Failure Responses. For example: 504, Gateway Time-out 6xx: Global Failure Responses. For example: 600, Busy Everywhere
	Reason-phrase	Textual description of the status code.

The traffic in an SIP-based communication system is routed through dedicated devices and applications (entities). In a multimedia communication session, these entities exchange messages.

One of the most common uses for SIP is VoIP, where SIP is used to set up the session. The following diagram illustrates how the messages and entities in a SIP-based communication system interoperate.

Figure 1. How SIP Works



The entity that initiates the call is referred to as the user agent (UA). The UA can be an SIP softphone (a PC-based application) or a SIP phone.

To initiate a call, the user agent sends an INVITE request to the previously configured SIP proxy server. The INVITE request contains the details of the destination, such as the destination uniform resource identifier (URI) and Call ID. In the diagram, the Caller A (user agent) sends an INVITE request to Proxy A.

When the proxy server receives the INVITE request, it sends a 100 (Trying) response to the user agent, Caller A. It also performs a DNS lookup to locate the SIP proxy server of the destination domain. After the SIP proxy server of the destination domain is located, the SIP proxy at the source domain sends the INVITE request to it. Here, Proxy A sends a 100 (Trying) response to Caller A and an INVITE request to Proxy B.

When the SIP proxy server of the destination domain receives the INVITE request from the SIP proxy server of the source domain, it responds with a 100 (Trying) response. It then sends the INVITE request to the destination user agent. In this

case, Proxy B sends a 100 (Trying) response to Proxy A and an INVITE request to Caller B.

When the destination user agent receives the INVITE request, it alerts Caller B and responds with a 180 (ringing) response. This response is routed back to the source user agent through the proxies.

When caller B accepts the call, the destination user agent responds with a 200 (OK) response. This signifies that caller B has answered the call. This response is routed back to the source user agent through the proxies. After the call is set up, the user agents communicate directly without the proxies.

The following table describes the entities of a SIP-based communication system and their roles.

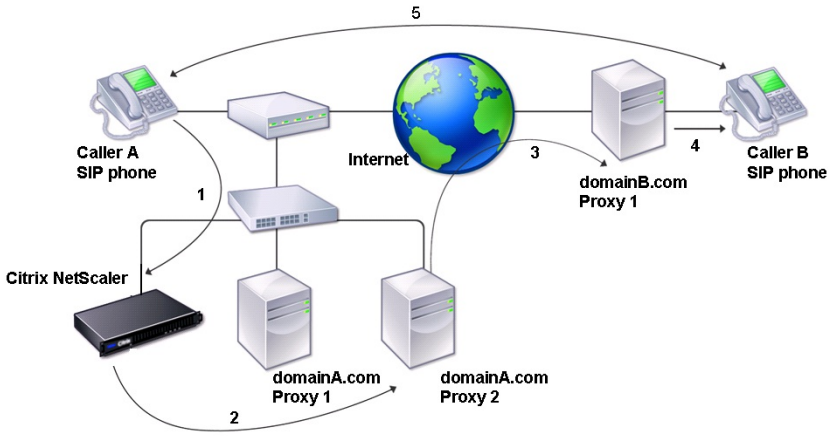
Entity	Role
User Agent (UA)	SIP user agents generate requests and respond to incoming requests. A user agent that generates requests is known as a User Agent Client (UAC). The user agent that responds to requests is known as the User Agent Server (UAS). In the preceding example, Caller A was the UAC and Caller B was the UAS.
Proxy Server	Proxies receive and route SIP requests based on the URI. They can selectively rewrite parts of the request message before forwarding it. They also handle registrations and invitations to user agents, and apply call policies.
Redirect Server	Redirect servers send routing information to the SIP proxy servers.
Registrar Server	Registrar servers provide location information to user agents and proxy servers.
Back-to-Back User Agent (B2BUA)	Back-to-Back User Agents (B2BUA) are combination of UAS and UAC.

You can configure the NetScaler appliance to load balance SIP requests to a group of SIP proxy servers. To do so, you need to create a load balancing virtual server with the load balancing method set to Call-ID hash, and then bind to it the services that are bound to the SIP proxies.

For load balancing to work, you must also configure the SIP proxies so that they do not add private IP addresses or private domains to the SIP header/payload. SIP proxies must add to the SIP header a domain name that resolves to the IP address of the SIP virtual server. Also, the SIP proxies must communicate with a common database to share registration information.

The NetScaler appliance can load balance SIP proxies in either a one-arm DSR configuration or an inline direct server return (DSR) configuration. In a one-arm DSR configuration, the appliance receives SIP requests from user agents and routes the requests to the appropriate SIP proxy by using the configured load balancing method. The SIP proxies send their responses to the destination SIP proxies, bypassing the appliance, as illustrated in the following diagram.

Figure 2. SIP in One-Arm Mode

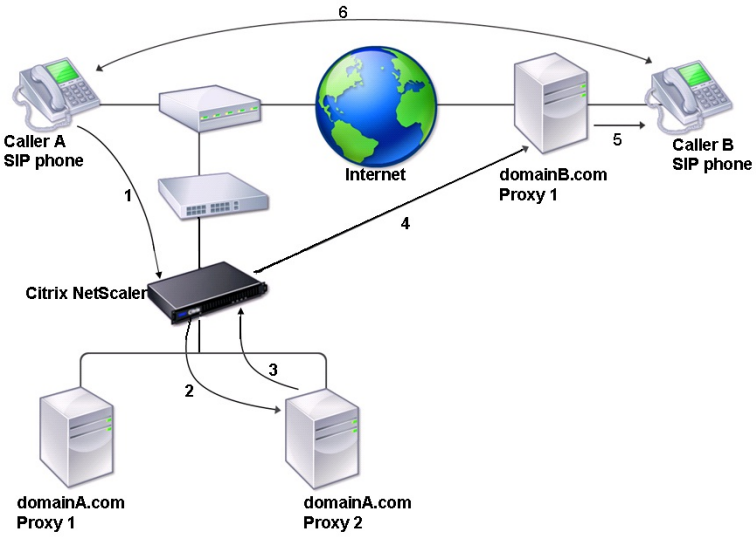


The flow of requests and responses in this configuration is as follows:

- The user agent, Caller A, sends an INVITE request to the NetScaler. The NetScaler, using a load balancing method, routes the request to Proxy 2.
- Proxy 2 receives the INVITE request from the NetScaler and responds with a 100 (Trying) message.
- Proxy 2 performs a DNS lookup to obtain the IP address of the destination SIP proxy at domainB.com. It then sends the INVITE request to the destination proxy.
- The destination proxy responds with a 100 (Trying) message and sends the INVITE request to the destination user agent, Caller B. The destination user agent, Caller B, begins to ring and responds with a 180 (Ringing) message. This message is sent to Caller A through the NetScaler and the Proxy 2. After the user accepts the call, Caller B responds with a 200 (OK) message that is propagated to Caller A through the NetScaler and the Proxy 2.
- After Caller B accepts the call, the user agents (Caller A and Caller B) communicate independently.

In an inline DSR configuration, the appliance is placed between the router and the SIP proxy, as illustrated in the following diagram.

Figure 3. SIP in Inline Mode



The flow of requests and responses is as follows:

- The user agent, Caller A, sends an INVITE request to the appliance. The NetScaler, using a load balancing method, routes the request to Proxy 2.
- Proxy 2 receives the INVITE request from the appliance and responds with a 100 (Trying) message.
- Proxy 2 performs a DNS lookup to obtain the IP address of the destination SIP proxy at domainb.com. It then propagates the INVITE request to the destination proxy through the appliance.
- The appliance performs RNAT, and replaces the source IP address in the INVITE request with the NAT IP address, and then forwards the INVITE request to the destination SIP proxy.
- The destination proxy responds with a 100 (Trying) message and sends the INVITE request to the destination user agent, Caller B. Caller B begins to ring and responds with a 180 (Ringing) message. This message is sent to Caller A through the NetScaler and the Proxy 2. After the user accepts the call, Caller B responds with a 200 (OK) message that is propagated to Caller A through the appliance and Proxy 2.
- After the user accepts the call, the user agents (Caller A and Caller B) communicate independently.

Parameter	Specifies
maxForwards	SIP packet max-forwards. Possible Values: 0-255. Default: 1.
sipMethod	SIP method to be used for the query. Possible values: OPTIONS, INVITE, REGISTER Default value: OPTIONS.
sipURI	SIP method string, sent to the server. For example “OPTIONS sip:sip.test.”
sipregURI	SIP user to be registered.

To configure built-in monitors to check the state of SIP server, see Configuring Monitors in a Load Balancing Setup.

Monitoring RADIUS Services

Aug 01, 2013

The NetScaler appliance RADIUS monitor periodically checks the state of the RADIUS service to which it is bound by sending an authentication request to the service. The RADIUS server authenticates the RADIUS monitor and sends a response. By default, the monitor expects to receive a response code of 2, the default Access-Accept response, from the RADIUS server. As long as the monitor receives the appropriate response, it marks the service UP.

Note: RADIUS monitor supports only PAP type authentication.

- If the client authenticated successfully, the RADIUS server sends an Access-Accept response. The default access-accept response code is 2, and this is the code that the appliance uses.
- If the client fails to authenticate successfully (such as when there is a mismatch in the user name, password, or secret key), the RADIUS server sends an Access-Reject response. The default access-reject response code is 3, and this is the code that the appliance uses.

Parameter	Specifies
userName	User name on the RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3 server. This user name is used in the probe.
password	Password used in monitoring RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3/LDAP servers.
radKey	Shared secret key value that the RADIUS server uses during client authentication.
radNASid	NAS-ID that is encapsulated in the payload when an access request is made.
radNASip	The IP address that is encapsulated in the payload when an access-request is made. When radNASip is not configured, the NetScaler sends the mapped IP address (MIP) to the RADIUS server as the NAS IP address.

To monitor a RADIUS service, you must configure the RADIUS server to which it is bound as follows:

1. Add the user name and password of the client that the monitor will use for authentication to the RADIUS authentication database.
2. Add the IP address and secret key of the client to the appropriate RADIUS database.
3. Add the IP addresses that the appliance uses to send RADIUS packets to the RADIUS database. If the NetScaler appliance has more than one mapped IP address, or if a subnet IP address (SNIP) is used, you must add the same secret key for all of the IP addresses.

   Caution: If the IP address used by the appliance are not added to the RADIUS database, the RADIUS server will discard all packets.

To configure built-in monitors to check the state of RADIUS server, see Configuring Monitors in a Load Balancing Setup.

Monitoring Accounting Information Delivery from a RADIUS Server

Sep 13, 2013

You can configure a monitor called a *RADIUS accounting* monitor to determine whether the Radius server used for Authentication, Authorization, and Accounting (AAA) is delivering accounting information as expected. The monitor is of type RADIUS_ACCOUNTING. The probe is generated by a Perl script called nsbmradius.pl, which is located in the /nsconfig/monitors/ directory. The script sends successive accounting request probes to the RADIUS server. The probe is considered successful only if the RADIUS accounting server responds with a packet whose Code field is set to 5, which, according to RFC 2866, indicates an Accounting-Response packet.

When configuring a RADIUS accounting monitor, you must specify a secret key. You can specify optional parameters, each of which represents a RADIUS attribute, such as Acct-Status-Type and Framed-IP-Address. For information about these attributes, see RFC 2865, "Remote Authentication Dial In User Service (RADIUS)," and RFC 2866, "RADIUS Accounting."

At the command prompt, type the following commands to configure a RADIUS accounting monitor and verify the configuration:

- add lb monitor <monitorName> RADIUS_ACCOUNTING [-userName <string>] {-password } {-radKey } [-radNASip <ip_addr>] [-radAccountType <positive_integer>] [-radFramedIP <ip_addr>] [-radAPN <string>] [-radMSISDN <string>] [-radAccountSession <string>]
- show lb monitor <monitorName>

Example

```
add lb monitor radAccntMon RADIUS_ACCOUNTING -radKey "8d#>9jr4rV)L7%a2-zW13sM"
```

Monitoring DNS and DNS-TCP Services

Mar 19, 2012

The NetScaler appliance has two built-in monitors that can be used to monitor DNS services: DNS and DNS-TCP. When bound to a service, either monitor periodically checks the state of that DNS service by sending a DNS query to it. The query resolves to an IPv4 or IPv6 address. That IP address is then checked against the list of test IP addresses that you configure. The list can contain up to five IP addresses. If the resolved IP address matches at least one IP address on the list, the DNS service is marked as up. If the resolved IP does not match any IP addresses on the list, the DNS service is marked as down.

Parameter	Parameter
query	The DNS query (domain name) sent to the DNS service that is being monitored. Default value: “\007” If the DNS query succeeds, the service is marked as UP; otherwise, it is marked as DOWN. For a reverse monitor, if the DNS query succeeds, the service is marked as DOWN; otherwise, it is marked as UP. If no response is received, the service is marked as DOWN.
queryType	The type of DNS query that is sent. Possible values: Address, Zone.
IPAddress	List of IP addresses that are checked against the response to the DNS monitoring probe.
IPv6	Select this check box if the IP address uses IPv6 format.

To configure the built-in DNS or DNS-TCP monitors, see Configuring Monitors in a Load Balancing Setup.

Monitoring LDAP Services

Mar 24, 2015

The NetScaler appliance has one built-in monitor that can be used to monitor LDAP services: the LDAP monitor. It periodically checks the LDAP service to which it is bound by authenticating and sending a search query to it. If the search is successful, the service is marked UP. If the LDAP server does not locate the entry, a failure message is sent to the LDAP monitor, and the service is marked DOWN.

You configure the LDAP monitor to define the search that it should perform when sending a query. You can use the Base DN parameter to specify a location in the directory hierarchy where the LDAP server should start the test query. You can use the Attribute parameter to specify an attribute of the target entity.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.

Parameter	Specifies
baseDN	Base name for the LDAP monitor from where the LDAP search must start. If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.
bindDN	BDN name for the LDAP monitor.
filter	Filter for the LDAP monitor.
password	Password used in monitoring LDAP servers.
attribute	Attribute for the LDAP monitor.

To configure the built-in LDAP monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring MySQL Services

Mar 24, 2015

The NetScaler appliance has one built-in monitor that can be used to monitor MySQL services: the MySQL monitor. It periodically checks the MySQL service to which it is bound by sending a search query to it. If the search is successful, the service is marked UP. If the MySQL server does not respond or the search fails, a failure message is sent to the MySQL monitor, and the service is marked DOWN.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.

Parameter	Specifies
database	Database that is used for the MySQL monitor.
sqlQuery	SQL query that is used for the MySQL monitor.

To configure built-in MySQL monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring SNMP Services

Mar 24, 2015

The NetScaler appliance has one built-in monitor that can be used to monitor SMNP services: the SNMP monitor. It periodically checks the SNMP agent on the service to which it is bound by sending a query for the enterprise identification ID (OID) that you configure for monitoring. If the query is successful, the service is marked UP. If the SNMP service finds the OID that you specified, the query succeeds and the SNMP monitor marks the service UP. If it does not find the OID, the query fails and the SNMP monitor marks service DOWN.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.

Parameter	Specifies
SNMPOID	OID that is used for the SNMP monitor.
snmpCommunity	Community that is used for the SNMP monitor.
snmpThreshold	Threshold that is used for the SNMP monitor.
snmpVersion	SNMP version that is used for load monitoring. Possible Values: V1, V2.

To configure the built-in SNMP monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring NNTP Services

Mar 24, 2015

The NetScaler appliance has one built-in monitor that can be used to monitor NNTP services: the NNTP monitor. It periodically checks the NNTP service to which it is bound by connecting to the service and checking for the existence of the newsgroup that you specify. If the newsgroup exists, the search is successful and the service is marked UP. If the NNTP service does not respond or the search fails, the service is marked DOWN.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.
The NNTP monitor can optionally be configured to post a test message to the newsgroup as well.

Parameter	Specifies
userName	User name on the RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3 server. This user name is used in the probe.
password	Password used in monitoring RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3/LDAP servers.
group	Group name to be queried for NNTP monitor.

To configure the built-in NNTP monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring POP3 Services

Mar 24, 2015

The NetScaler appliance has one built-in monitor that can be used to monitor POP3 services: the POP3 monitor. It periodically checks the POP3 service to which it is bound by opening a connection with a POP3 server. If the POP3 server responds with the correct response codes within the configured time period, it marks the service UP. If the POP3 service does not respond, or responds incorrectly, it marks the service DOWN.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.

Parameter	Specifies
userName	User name POP3 server. This user name is used in the probe.
password	Password used in monitoring POP3 servers.
scriptName	The path and name of the script to execute.
dispatcherIP	The IP address of the dispatcher to which the probe is sent.
dispatcherPort	The port of the dispatcher to which the probe is sent.

To configure the built-in POP3 monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring SMTP Services

Mar 24, 2015

The NetScaler appliance has one built-in monitor that can be used to monitor SMTP services: the SMTP monitor. It periodically checks the SMTP service to which it is bound by opening a connection with it and conducting a series of handshakes to ensure that the server is operating correctly. If the SMTP service completes the handshakes properly, the monitor marks the service UP. If the SMTP service does not respond, or responds incorrectly, it marks the service DOWN.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.

Parameter	Specifies
userName	User name SMTP server. This user name is used in the probe.
password	Password used in monitoring SMTP servers.
scriptName	The path and name of the script to execute.
dispatcherIP	The IP Address of the dispatcher to which the probe is sent.
dispatcherPort	The port of the dispatcher to which the probe is sent.

To configure the built-in SMTP monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring RTSP Servers

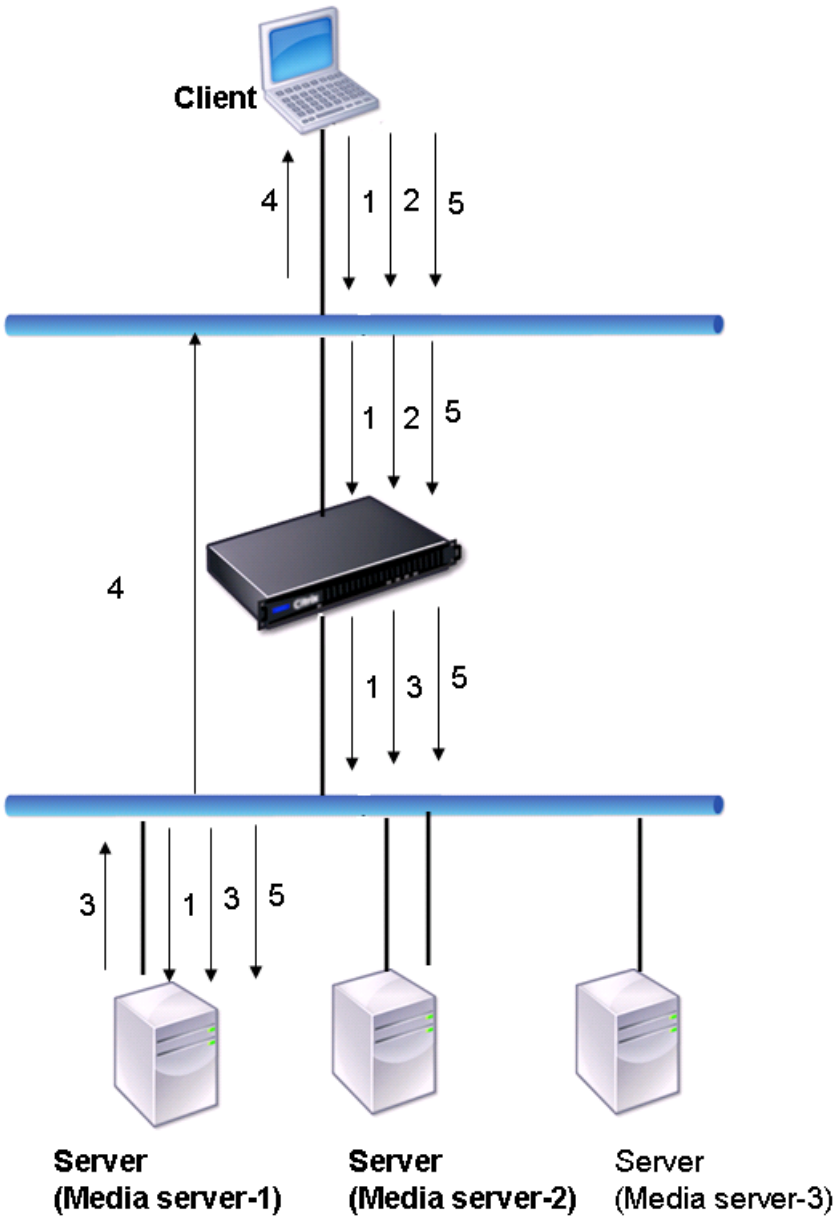
Apr 08, 2013

The NetScaler appliance has one built-in monitor that can be used to monitor RTSP services: the RTSP monitor. It periodically checks the RTSP service to which it is bound by opening a connection with the load balanced RTSP server. The type of connection that it opens, and the response that it expects, differs depending upon the network configuration. If the RTSP service responds as expected within the configured time period, it marks the service UP. If the service does not respond, or responds incorrectly, it marks the service DOWN.

The NetScaler appliance can be configured to load balance RTSP servers using two topologies: NAT-off and NAT-on. RTSP servers send their responses directly to the client, bypassing the appliance. The appliance must be configured to monitor RTSP services differently depending upon which topology your network uses. The appliance can be deployed either in inline or non-inline mode in both NAT-off and NAT-on mode.

In NAT-off mode, the appliance operates as a router: it receives RTSP requests from the client and routes them to the service that it selects using the configured load balancing method. If your load balanced RTSP servers are assigned publicly accessible FQDNs in DNS, the load balanced servers send their responses directly to the client, bypassing the appliance. The following figure demonstrates this configuration.

Figure 1. RTSP in NAT-off Mode



The flow of requests and responses in this scenario is as follows:

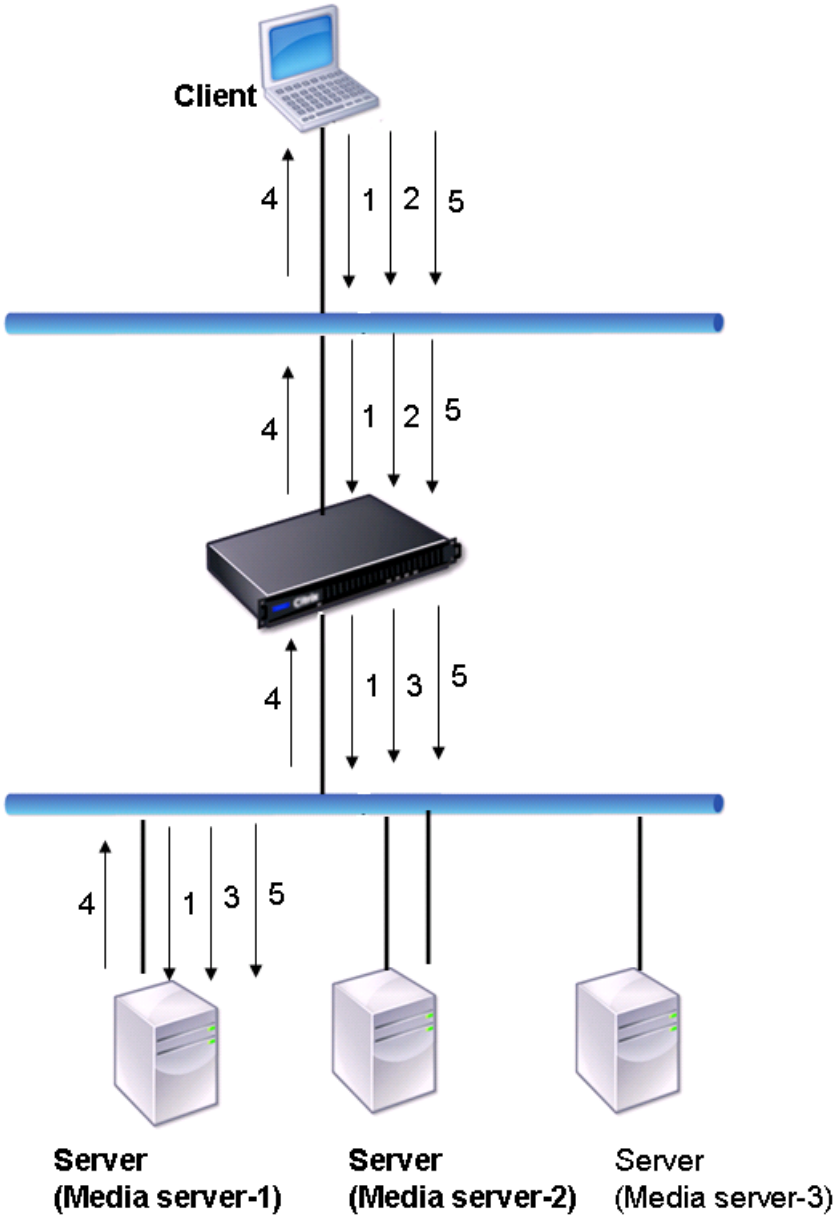
1. The client sends a DESCRIBE request to the appliance. The appliance uses the configured load balancing method to choose a service, and routes the request to Media Server-1.
2. The client sends a SETUP request to the appliance. If the RTSP session ID is exchanged in the DESCRIBE request, the appliance, using RTSPSID persistence, routes the request to Media Server-1. If the RTSP session ID is exchanged in the SETUP request, the appliance does one of the following:
   - If the RTSP request comes on the same TCP connection, it routes the request to Media Server-1, maintaining persistence.
   - If the request arrives on a different TCP connection, it uses the configured load balancing method to choose a service, and sends the request to that service, not maintaining persistence. This means that the request may be sent to a different service.
3. Media Server-1 receives the SETUP request from the appliance, allocates resources to process the RTSP request, and sends the appropriate session ID to the client.

Note: The appliance does not perform NAT to identify the RTSP connection, because the RTSP connections bypass it.

4. For subsequent requests, the client then uses the session ID to identify the session and send control messages to the media server. Media Server-1 performs the requested actions, such as play, forward, or rewind.

In NAT-on mode, the appliance receives RTSP requests from the client and routes those requests to the appropriate media server using the configured load balancing method. The media server then sends its responses to the client through the appliance, as illustrated in the following diagram.

Figure 2. RTSP in NAT-on Mode



The flow of requests and responses in this scenario is as follows:

1. The client sends a DESCRIBE request to the appliance. The appliance uses the configured load balancing method to choose a service, and routes the request to Media Server-1.
2. The client sends a SETUP request to the appliance. If the RTSP session ID is exchanged in the DESCRIBE request, the appliance, using the RTSPSID persistence, routes the request to Media Server-1. If the RTSP session ID is exchanged in the SETUP request, the appliance does one of the following:

- If the RTSP request comes on the same TCP connection, it routes the request to Media Server-1, maintaining persistence.
- If the request arrives on a different TCP connection, it uses the configured load balancing method to choose a service, and sends the request to that service, not maintaining persistence. This means that the request may be sent to a different service.

3. Media Server-1 receives the SETUP request from the appliance, allocates resources to process the RTSP request, and sends the appropriate session ID to the client.
4. The appliance performs NAT to identify the client for RTSP data connections, and the RTSP connections pass through the appliance and are routed to the correct client.
5. For subsequent requests, the client then uses the session ID to identify the session and send control messages to the appliance. The appliance uses RTSPSID persistence to identify the appropriate service, and routes the request to Media Server-1. Media Server-1 performs the requested action, such as play, forward, or rewind.

The RTSP monitor uses the RTSP protocol to evaluate the state of the RTSP services. The RTSP monitor connects to the RTSP server and conducts a sequence of handshakes to ensure that the server is operating correctly.

Parameter	Specifies
rtspRequest	The RTSP request string that is sent to the RTSP server (for example, OPTIONS *). The default value is 07. The length of the request must not exceed 163 characters.
respCode	Set of response codes that are expected from the service.

For instructions on configuring an RTSP monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring the XML Broker Services

Sep 13, 2013

The NetScaler appliance has a built-in monitor type, CITRIX-XML-SERVICE, with which you can create monitors to monitor the XML Broker services. The XML Broker services are used by Citrix XenApp. The monitor opens a connection to the service and periodically probes the XML services to which it is bound. If the server responds as expected within the configured time period, the monitor marks the service UP. If the service does not respond, or responds incorrectly, the monitor marks the service DOWN.

To configure a CITRIX-XML-SERVICE monitor, you need to specify the application name in addition to setting the standard parameters. The application name is the name of the application that has to be run to monitor the state of the XML Broker service. The default application is Notepad.

To configure monitors for XML Broker services, see "Configuring Monitors in a Load Balancing Setup."

Monitoring ARP Requests

Mar 19, 2012

The NetScaler appliance has one built-in monitor that can be used to monitor ARP requests: the ARP monitor. This monitor periodically sends an ARP request to the service to which it is bound, and listens for the expected response. If it receives the expected response, it marks the service UP. If it receives no response or the wrong response, it marks the service DOWN.

ARP locates a hardware address for a load balanced server when only the network layer address is known. ARP works with IPv4 to translate IP addresses to Ethernet MAC addresses. ARP monitoring is not relevant to IPv6 networks, and is therefore not supported on those networks.

There are no special parameters for the ARP monitor.

For instructions on configuring an ARP monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring the NetScaler Gateway

Mar 19, 2012

The NetScaler appliance has one built-in monitor that can be used to monitor a load-balanced NetScaler Gateway: the CITRIX-AG monitor. This is in addition to two monitors for the Advanced Access Control login page and agent service page, which are described separately. The CITRIX-AG monitor periodically logs on to the NetScaler Gateway service to which it is bound, and awaits the expected responses to its requests. If it receives the expected responses, it marks the service UP. If it receives no response or the wrong responses, it marks the service DOWN.

To configure monitoring of an NetScaler Gateway, you must first create a local user and password for the monitor on the load balanced NetScaler Gateway server that the service is bound to. After you configure the NetScaler Gateway, you then configure the monitor. The monitor logs on to the NetScaler Gateway using the realm and user name. For example, if you configured a realm of LDAP and a user name of user1, the NetScaler Gateway logs on as LDAP/user1.

Note: RSA SecurID authentication is not supported for this monitor. RSA SecurID requires an RSA-generated token as a password, which is not supported on the NetScaler appliance.

Parameter	Specifies
userName	A user name.
password	A password for the username.
secondaryPassword	A secondary password for the username.

For instructions on configuring the CITRIX-AG monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring the Advanced Access Control Login Page

Jul 10, 2013

The NetScaler appliance has one built-in monitor that can be used to monitor the Advanced Access Control (AAC) login page on a load-balanced NetScaler Gateway: the CITRIX-AAC-LOGINPAGE monitor. This monitor periodically logs on to the AAC login page via the NetScaler Gateway service to which it is bound, and awaits the expected response. If it receives the expected response, it marks the service UP. If it receives no response or the wrong response, it marks the service DOWN.

Parameter	Specifies
logonpointName	The URL from which users access corporate resources using the NetScaler Gateway Advanced Edition. This setting controls access to server farms, the Access Interface configuration, and other session-specific settings. It can also be used as a filter within NetScaler Gateway policies.

For instructions on configuring the CITRIX-AAC-LOGINPAGE monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring the Advanced Access Control Logon Agent Service Page

Jul 10, 2013

The NetScaler appliance has one built-in monitor that can be used to monitor the Advanced Access Control (AAC) agent service page on a load-balanced NetScaler Gateway: the CITRIX-AAC-LAS monitor. The Logon Agent Service (LAS) is a service component of Advanced Access Control that requests authentication to the Authentication Service. This monitor periodically logs on to the AAC agent service page via the NetScaler Gateway service to which it is bound, and awaits the expected response. If it receives the expected response, it marks the service UP. If it receives no response or the wrong response, it marks the service DOWN.

Parameter	Specifies
logonpointName	The URL from which users access corporate resources using the NetScaler Gateway advanced edition. This setting controls access to server farms, the Access Interface configuration, and other session-specific settings. It can also be used as a filter within NetScaler Gateway policies.
lasVersion	The version number of the agent.

For instructions on configuring the CITRIX-AAC-LAS monitor, see Configuring Monitors in a Load Balancing Setup.

Monitoring the XenDesktop Delivery Controller Services

Nov 26, 2013

In desktop virtualization, the NetScaler appliance can be used to load balance the Web Interface (WI) servers and the XenDesktop Delivery Controller servers deployed by Citrix XenDesktop environment. The NetScaler provides a built-in monitor, CITRIX-XD-DDC monitor, which monitors the XenDesktop Delivery Controller servers. In addition to the health check, you can also verify whether the probe is sent by a valid user of the XenDesktop Delivery Controller server.

The monitor sends a probe to the XenDesktop Delivery Controller server in the form of an XML message. If the server responds to the probe with the identity of the server farm, the probe is considered to be successful and the server's status is marked as UP. If the HTTP response does not have a success code or the identity of the server farm is not present in the response, the probe is considered to be a failure and the server's status is marked as DOWN.

The Validate Credentials option determines the probe to be sent by the monitor to the XenDesktop Delivery Controller server, that is, whether to request only the server name or to also validate the login credentials.

Note: Regardless of whether or not the user credentials (user name, password and domain) are specified on the CITRIX-XD-DDC monitor, the XenDesktop Delivery Controller server validates the user credentials only if the option to validate credentials is enabled on the monitor.
If you use the wizard for configuring the load balancing of the XenDesktop servers, the CITRIX-XD-DDC monitor is automatically created and bound to the XenDesktop Delivery Controller services. If you do not use the wizard, add a monitor of the type CITRIX-XD-DDC.

- For instructions on using the wizard, see Configuring the load balancing of XenDesktop.
- For instructions on adding a monitor, see Creating Monitors.
- For instructions on binding a monitor to a service, see Binding Monitors to Services.

At the command prompt, type the following commands to add an XD-DDC monitor and verify the configuration:

- add lb monitor <monitorName> <monitorType> -userName <userName> -password <password> -ddcDomain <ddc_domain_name> -validateCred YES
- show lb monitor <monitorName>

Example

```
 > add lb monitor xdddcmon Citrix-xd-ddc -userName Administrator -password E12Dc35450a1 -ddcDomain dhop -validateCred YES
 Done
> show lb monitor xdddcmon
1) Name......:xdddcmon Type......:CITRIX-XD-DDC State......: ENABLED

Standard parameters:
Interval......:..5 sec...Retries......:..3
Response timeout......:..2 sec...Down time......:..30 sec
Reverse......:..NO...Transparent......:..NO
Secure......:..NO...LRTM......:..ENABLED
Action......:..Not applicable...Deviation......:..0 sec
Destination IP......:..Bound service
Destination port......:..Bound service
Iptunnel......:..NO
TOS.....:..NO...TOS ID.....:..0
SNMP Alert Retries.....:..0...Success Retries.....:..1
Failure Retries.....:..0

Special parameters:
User Name..........:"Administrator"
Password..........:*****
DDC Domain.........: "dhop"
Done
```

At the command prompt, type:

set lb monitor <monitorName> <monitorType> -userName -password -ddcDomain <ddc_domain_name> -validateCred YES

Example

```
 > set lb monitor XD_DDC_21.21.21.22_443_mn CITRIX-xd-ddc -userName Administrator -password D123S1R2A123 -ddcDomain dhop -validateCred YES
Done
```

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, do one of the following:
   - To add an XD-DDC monitor, click Add.
   - To modify an XD-DDC monitor, select the monitor, and click Open.
3. Type a name for the monitor.

4. Select the monitor type as CITRIX-XD-DDC.
5. On the Special Parameters tab:
   - Validate Credentials—validateCred (To specify YES, select the check box.)
   - Name*—monitorName
   - Type*—monitorType
   - User Name*—userName
   - Password*—password
   - Domain Name*—ddcDomain

   *A required parameter
6. Click Create.
7. Select the new monitor, click Open, and verify the settings.

Monitoring Web Interface Services

May 07, 2015

In desktop virtualization, the NetScaler appliance can be used to load balance the Web Interface (WI) servers and Dynamic Desktop Controller (DDC) servers deployed in the Citrix XenApp and Citrix XenDesktop and environments. The NetScaler appliance has two built-in monitor types for monitoring the WI servers used in these environments.

A CITRIX-WEB-INTERFACE monitor can monitor the Web Interface services efficiently because it monitors a dynamic page at the location specified by the site path. The monitor checks for critical failures in resource availability.

To mark a service as UP, the appliance expects the following response from the server:

1. For the first GET request, 200 OK .
2. For the POST request with credentials, 302 Found with the required WIAuthID.
3. For the last GET request with session cookie, 200 OK.

Note: If a redirect URL is configured, 302 Found is expected in the first request before 200 OK.
Note: Monitor probes originate from the NetScaler IP (NSIP) address.
When you configure a CITRIX-WEB-INTERFACE monitor, specify the site path to the location of the http page that displays the data collected by the monitor. To monitor the status of the service, in the specified site path, you can view the data updated dynamically by the monitoring script auth/nocookies.aspx.

Note: End the site path with a slash (/) to indicate that the monitored resource is dynamic.
Note: When you configure the WI-EXTENDED monitor, when specifying the site path, do not enter a slash (/) at the end of the path as the software internally adds a slash at the end of the path. For example, note the following command:

```
add monitor wi CITRIX-WI-EXTENDED -sitepath "/Citrix/DesktopWeb" -username aaa -password bbb -domain ccc
```

A CITRIX-WI-EXTENDED monitor verifies the logging process with the Web Interface service. This monitor accesses the login page and passes the user name, password, domain, and site path that were specified while configuring the monitor. It verifies the validity of the login credentials, correct configuration of the monitor (for example, the site path), and the connection with the IIS server.

Note: The CITRIX-WI-EXTENDED monitor is supported only for the .NET version of the WI servers. This monitor will not work for the JSP version of the WI servers.
If you use the wizard for configuring load balancing of the XenDesktop servers, a CITRIX-WEB-INTERFACE monitor is automatically created and bound to the WI services. The wizard adds and binds a CITRIX-WEB-INTERFACE monitor by default. If you want to add and bind a CITRIX-WI-EXTENDED monitor, select the Validate Credentials check box and type the necessary data. If you do not use the wizard, add a monitor corresponding to the WI services and bind it to each WI service that you create.

- For instructions on using the wizard, see Configuring XenDesktop for Load Balancing or Configuring XenApp for Load Balancing.
- For instructions on adding a CITRIX-WEB-INTERFACE monitor, see Creating Monitors.
- For instructions on binding a monitor to a service, see Binding Monitors to Services.

At the command prompt, type:

add lb monitor <monitorName> <monitorType> -sitePath <site_path> -dispatcherIP 127.0.0.1 -dispatcherPort 3013 -userName <username> -password <password> -domain <domain_name>

Examples

```
add lb monitor mwie CITRIX-WEB-INTERFACE -sitePath "/Citrix/XDWI/"
add lb monitor mwie CITRIX-WI-EXTENDED -sitePath "/Citrix/XDWI/"
-dispatcherIP 127.0.0.1 -dispatcherPort 3013 -userName administrator
```

```
-password d83d154575d426 -encrypted -domain wi
```

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, do one of the following:
   - To add a WI monitor, click Add.
   - To modify a WI monitor, select the monitor, and click Open.
3. Type a name for the monitor.
4. Select the monitor type as CITRIX-WEB-INTERFACE or CITRIX-WI-EXTENDED.
5. On the Special Parameters tab, type the site path. To configure the CITRIX-WI-EXTENDED monitor, specify values for the following parameters:
   - User Name*—userName
   - Password*—password
   - Domain Name*—domain
6. Click Create.
7. Select the new monitor, click Open, and verify the settings.

Monitoring Citrix StoreFront Stores

Sep 30, 2015

You can configure a user monitor for a Citrix Storefront store. The monitor determines the state of the StoreFront store by successively probing the account service, authentication service, and discovery document (in that order). If any of those services do not respond to the probe, the monitor probe fails, and the StoreFront store is marked as DOWN. The monitor sends probes to the IP address and port of the bound service.

Note: Monitor probes originate from the NetScaler IP (NSIP) address. However, if the subnet of a StoreFront server is different from that of the appliance, then the subnet IP (SNIP) address is used.
Beginning with build 120.13, you can also bind a StoreFront monitor to a service group. A monitor is bound to each member of the service group and probes are sent to the IP address and port of the bound member (service). Also, because each member of a service group is now monitored by using the member's IP address, you can now use the StoreFront monitor to monitor StoreFront cluster nodes that are added as members of the service group.

The hostname parameter for StoreFront monitors is deprecated. The secure parameter is now used to determine whether to use HTTP (the default) or HTTPS to send monitor probes.

To use HTTPS, set the secure option to Yes.

At the command prompt, type the following commands to configure a StoreFront monitor and verify the configuration:

- add lb monitor <monitorName> STOREFRONT <string> -storeName <string> [-storefrontacctservice (YES | NO)] -secure (YES | NO)
- show lb monitor <monitorName>

Example

```
add lb monitor storefront_ssl STOREFRONT -storename myStore -storefrontacctservice YES -secure YES
```

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the Create Monitor dialog box, in the details pane, click Add, and then, in the Create Monitor dialog box, in the Type list, select STOREFRONT.
3. On the Standard Parameters tab, to use HTTPS, select Secure.
4. On the Special Parameters tab, set the following parameters:
   - Store Name
   - StoreFront Account Service
5. Click OK.

Custom Monitors

Jun 09, 2015

In addition to built-in monitors, you can use custom monitors to check the state of your services. The NetScaler appliance provides several types of custom monitors based on scripts that are included with NetScaler operating system that can be used to determine the state of services based on the load on the service or network traffic sent to the service. These are the inline monitors, user monitors, and load monitors.

With any of these types of monitors, you can use the supplied functionality, or you can create your own scripts and use those scripts to determine the state of the service to which the monitor is bound.

This section includes the following details:

- Configuring HTTP-Inline Monitors
- Understanding User Monitors
- How to Use a User Monitor to Check Web Sites
- Understanding the Internal Dispatcher
- Configuring a Custom User Monitor
- Understanding Load Monitors
- Configuring Load Monitors
- Unbinding Metrics from a Metrics Table

Configuring HTTP-Inline Monitors

Apr 15, 2015

Inline monitors analyze and probe the responses from the services to which they are bound only when those services receive client requests. The inline monitor is of type HTTP-INLINE and can only be configured to work with HTTP and HTTPS services. An inline monitor determines that the service to which it is bound is UP by checking its responses to the requests that are sent to it. When no client requests are sent to the service, the inline monitor probes the service by using the configured URL.

Note: Inline monitors cannot be bound to HTTP or HTTPS Global Server Load Balancing (GSLB) remote or local services because these services represent virtual servers rather than actual load balanced Web servers.
Inline monitors have a time-out value and a retry count when probes fail. You can select any of the following action types for the NetScaler appliance to take when a failure occurs:

- **NONE.** No explicit action is taken. You can view the service and monitor, and the monitor indicates the number of current contiguous error responses and cumulative responses checked.
- **LOG.** Logs the event in ns/syslog and displays the counters.
- **DOWN.** Marks the service down and does not direct any traffic to the service. This setting breaks any persistent connections to the service. This action also logs the event and displays counters.

After the service is down, the service remains DOWN for the configured down time. After the DOWN time elapses, the inline monitor uses the configured URL to probe the service to see if it is available again. If the probe succeeds, the state of the service is changed to UP. Traffic is directed to the service, and monitoring resumes as before.

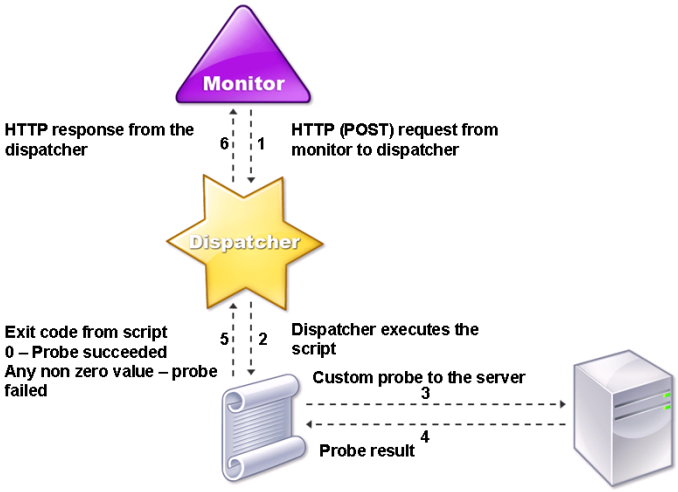
To configure inline monitors, see Configuring Monitors in a Load Balancing Setup.

Understanding User Monitors

Sep 09, 2015

User monitors extend the scope of custom monitors. You can create user monitors to track the health of customized applications and protocols that the NetScaler appliance does not support. The following diagram illustrates how a user monitor works.

Figure 1. User Monitors



A user monitor requires the following components.

- **Dispatcher**. A process, on the appliance, that listens to monitoring requests. A dispatcher can be on the loopback IP address (127.0.0.1) and port 3013. Dispatchers are also known as internal dispatchers. A dispatcher can also be a web server that supports Common Gateway Interface (CGI). Such dispatchers are also known as external dispatchers. They are used for custom scripts that do not run on the FreeBSD environment, such as .NET scripts.

  Note: You can configure the monitor and the dispatcher to use HTTPS instead of HTTP by enabling the “secure” option on the monitor and configure it as an external dispatcher. However, an internal dispatcher understands only HTTP, and cannot use HTTPS.

  In a HA setup, the dispatcher runs on both the primary and secondary NetScaler appliances. The dispatcher remains inactive on the secondary appliance.
- **Script**. The script is a program that sends custom probes to the load balanced server and returns the response code to the dispatcher. The script can return any value to the dispatcher, but if a probe succeeds, the script must return a value of zero (0). The dispatcher considers any other value as probe failure.

  The NetScaler appliance is bundled with sample scripts for commonly used protocols. The scripts exist in the /nsconfig/monitors directory. If you want to add a new script, add it there. If you want to customize an existing script, create a copy with a new name and modify it.

Important: Starting with release 10.1 build 122.17, the script files for user monitors are in a new location. If you upgrade an

MPX or VPX virtual appliance to release 10.1 build 122.17 or later, the changes are as follows:

- A new directory named conflicts is created in /nsconfig/monitors/ and all the built-in scripts of the previous builds are moved to this directory.
- All new built-in scripts are available in the /netscaler/monitors/ directory. All custom scripts are available in the /nsconfig/monitors/ directory.
- You must save a new custom script in the /nsconfig/monitors/ directory.
- After the upgrade is completed, if a custom script is created and saved in the /nsconfig/monitors/ directory, with the same name as that of a built-in script, the script in the /netscaler/monitors/ directory takes priority. That is, the custom script does not run.

If you provision a virtual appliance with release 10.1 build 122.17 or later, the changes are as follows:

- All built-in scripts are available in the /netscaler/monitors/ directory.
- The /nsconfig/monitors/ directory is empty.
- If you create a new custom script, you must save it in the /nsconfig/monitors/ directory.

For the scripts to function correctly, the name of the script file must not exceed 63 characters, and the maximum number of script arguments is 512. To debug the script, you must run it by using the nsumon-debug.pl script from the NetScaler command line. You use the script name (with its arguments), IP address, and the port as the arguments of the nsumon-debug.pl script. Users must use the script name, IP address, port, time-out, and the script arguments for the nsumon-debug.pl script.

Important: Starting with release 10.1 build 133.x, script files for user monitors support IPv6 addresses and include the following changes:

- For the following protocols, new pm files have been included for IPv6 support.
  - Radius
  - NNTP
  - POP3
  - SMTP
- The following sample scripts in /netscaler/monitors/ has been updated for IPv6 support:
  - nsbmradius.pl
  - nsldap.pl
  - nsnntp.pl
  - nspop3 nssf.pl
  - nssnmp.pl
  - nswi.pl
  - nstftp.pl
  - nssmtp.pl
  - nsrdp.pl
  - nsntlm-lwp.pl
  - nsftp.pl
  - nsappc.pl
- The following LB monitor types have been updated to support IPv6 addresses:
  - SMTP
  - NNTP
  - LDAP

- SNMP
- POP3
- FTP_EXTENDED (If you need to monitor the FTP services with IPv6 addresses, use FTP_EXTENDED monitor instead of USER monitor that uses nsftp.pl script.)

  Note: If the configured monitor is of type USER, you do not have to make any changes to your existing custom scripts.

To track the status of the server, the monitor sends an HTTP POST request to the configured dispatcher. This POST request contains the IP address and port of the server, and the script that must be executed. The dispatcher executes the script as a child process, with user-defined parameters (if any). Then, the script sends a probe to the server. The script sends the status of the probe (response code) to the dispatcher. The dispatcher converts the response code to an HTTP response and sends it to the monitor. Based on the HTTP response, the monitor marks the service as up or down.

The appliance logs the error messages to the /var/nslog/nsumond.log file when user monitor probes fail. The following table lists the user monitors and the possible reasons for failure.

User monitor type	Probe failure reasons
SMTP	Monitor fails to establish a connection to the server.
NNTP	Monitor fails to establish a connection to the server.
	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.
	Monitor fails to find the NNTP group.
LDAP	Monitor fails to establish a connection to the server.
	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.
	Monitor fails to bind to the LDAP server.
	Monitor fails to locate an entry for the target entity in the LDAP server.
FTP	The connection to the server times out.
	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.

User monitor type	Probe failure reasons
	Logon fails.
	Monitor fails to find the file on the server.
POP3	Monitor fails to establish a connection to the database.
	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.
	Logon fails.
POP3	Monitor fails to establish a connection to the database.
	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.
	Logon fails.
	Preparation of SQL query fails.
	Execution of SQL query fails.
SNMP	Monitor fails to establish a connection to the database.
	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.
	Logon fails.
	Monitor fails to create the SNMP session.
	Monitor fails to find the object identifier.
	The monitor threshold value setting is greater than or equal to the actual threshold of the monitor.
RDP (Windows Terminal Server)	Missing or invalid script arguments, which can include an invalid number of arguments or argument format.

User monitor type	Probe failure reasons
	Monitor fails to create a socket.
	Mismatch in versions.
	Monitor fails to confirm connection.

You can view the log file from the NetScaler command line by using the following commands, which open a BSD shell, display the log file on the screen, and then close the BSD shell and return you to the NetScaler command prompt:

```
> shell
root@ns# cat /var/nslog/nsumond.log
root@ns# exit
>
```

User monitors also have a time-out value and a retry count for probe failures. You can use user monitors with non-user monitors. During high CPU utilization, a non-user monitor enables faster detection of a server failure.

Note: If the user monitor probe times out during high CPU usage, the state of the service remains unchanged.

How to Use a User Monitor to Check Web Sites

Mar 19, 2012

You can configure a user monitor to check for specific Web site problems that are reported by HTTP servers using specific HTTP codes. The following table lists the HTTP response codes that this user monitor expects.

HTTP response code	Meaning
200 - success	Probe success.
503 - service unavailable	Probe failure.
404 - not found	Script not found or cannot execute.
500 - Internal server error	Internal error/resource constraints in dispatcher (out of memory, too many connections, unexpected system error, or too many processes). The service is not marked DOWN.
400 - bad request	Error parsing HTTP request.
502 - bad gateway	Error decoding script's response.

You configure the user monitor for HTTP by using the following parameters.

Parameter	Specifies
scriptName	The path and name of the script to execute.
scriptArgs	The strings that are added in the POST data. They are copied to the request verbatim.
dispatcherIP	The IP address of the dispatcher to which the probe is sent.
dispatcherPort	The port of the dispatcher to which the probe is sent.
localfileName	The name of a monitor script file on the local system.

Parameter	Specifies
destPath	A particular location on the NetScaler appliance where the uploaded local file is stored.

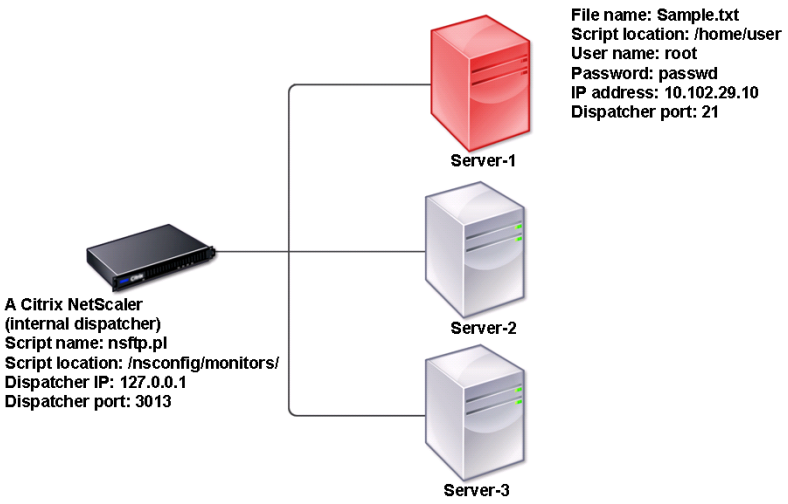
To create a user monitor to monitor HTTP, see Configuring Monitors in a Load Balancing Setup.

Understanding the Internal Dispatcher

Mar 19, 2012

You can use a custom user monitor with the internal dispatcher. Consider a case where you need to track the health of a server based on the presence of a file on the server. The following diagram illustrates this scenario.

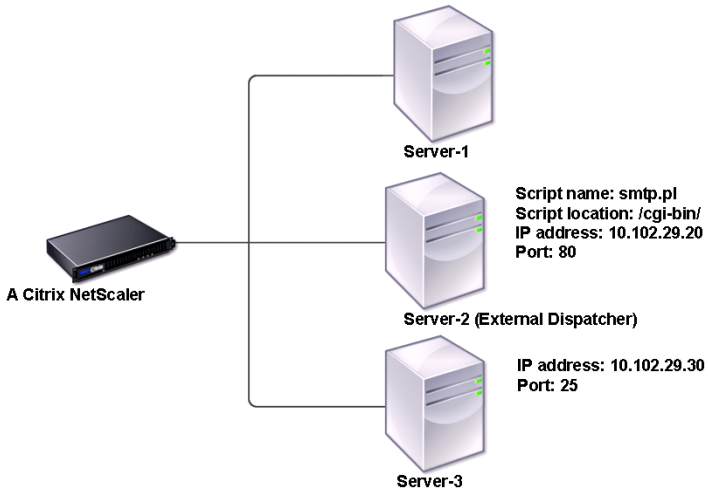
Figure 1. Using a User Monitor with the Internal Dispatcher



A possible solution is to use a Perl script that initiates an FTP session with the server and checks for the presence of the file. You can then create a user monitor that uses the Perl script. The NetScaler includes such a Perl script (nsftp.pl), in the /nsconfig/monitors/ directory.

You can use a user monitor with an external dispatcher. Consider a case where you must track the health of a server based on the state of an SMTP service on another server. This scenario is illustrated in the following diagram.

Figure 2. Using a User Monitor with an External Dispatcher



A possible solution would be to create a Perl script that checks the state of the SMTP service on the server. You can then create a user monitor that uses the Perl script.

Configuring a Custom User Monitor

Aug 24, 2016

To configure a custom user monitor, you must first write the script that performs the action that the monitor will use to check the service that is bound to it, and upload the script to the /nsconfig/monitors directory on the NetScaler appliance. Then you create the monitor on the appliance, as described below.

Note: Monitor probes originate from the NetScaler IP (NSIP) address.

To configure a user monitor by using the command line interface

At the command prompt, type:

add lb monitor <monitorName> USER -scriptname <NameOfScript> –scriptargs <Arguments>

Example

```
add monitor Monitor-User-1 USER -scriptname nsftp.pl –scriptargs "file=/home/user/
sample.txt;user=root;password=passwd"
```

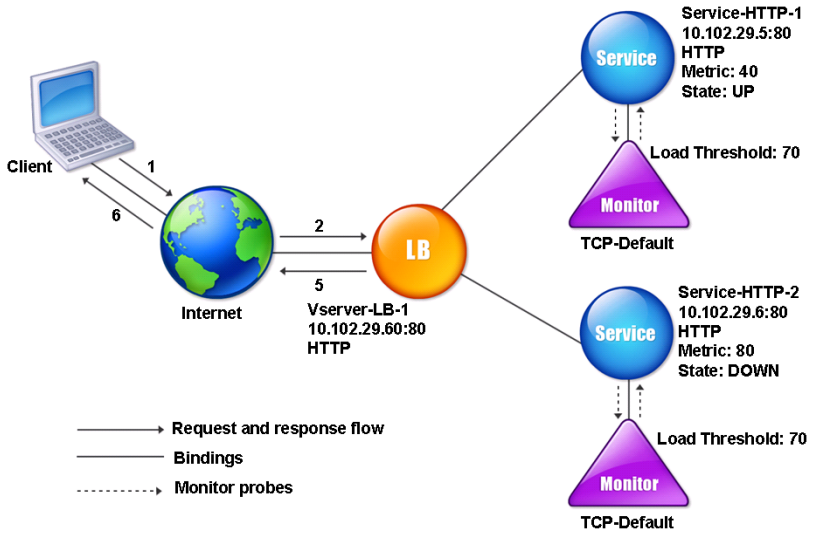
Understanding Load Monitors

Mar 19, 2012

Load monitors use SNMP polled OIDs to calculate load. The load monitor uses the IP address of the service to which it is bound (the destination IP address) for polling. It sends an SNMP query to the service, specifying the OID for a metric. The metrics can be CPU, memory, or number of server connections. The server responds to the query with a metric value. The metric value in the response is compared with the threshold value. The NetScaler appliance considers the service for load balancing only if the metric is less than the threshold value. The service with the lowest load value is considered first.

The following diagram illustrates a load monitor configured for the services described in the basic load balancing setup discussed in Setting Up Basic Load Balancing.

Figure 1. Operation of Load Monitors



Note: The load monitor does not determine the state of the service. It only enables the appliance to consider the service for load balancing.

After you configure the load monitor, you must then configure the metrics that the monitor will use. For load assessment, the load monitor considers server parameters known as metrics, which are defined within the metric tables in the appliance configuration. Metric tables can be of two types:

- **Local.** By default, this table exists in the appliance. It consists of four metrics: connections, packets, response time, and bandwidth. The appliance specifies these metrics for a service, and SNMP queries are not originated for these services. These metrics cannot be changed.
- **Custom.** A user-defined table. Each metric is associated with an OID.

By default, the appliance generates the following tables:

- NetScaler
- RADWARE
- CISCO-CSS
- LOCAL

- FOUNDRY
- ALTEON

You can either add the appliance-generated metric tables, or you can add tables of your own choosing, as shown in the following table. The values in the metric table are provided only as examples. In an actual scenario, consider the real values for the metrics.

Metric name	OIDs	Weight	Threshold
CPU	1.2.3.4	2	70
Memory	4.5.6.7	3	80
Connections	5.6.7.8	4	90

To calculate the load for one or more metrics, you assign a weight to each metric. The default weight is 1. The weight represents the priority given to each metric. If the weight is high, the priority is high. The appliance chooses a service based on the SOURCEIPDESTIP hash algorithm.

You can also set the threshold value for each metric. The threshold value enables the appliance to select a service for load balancing if the metric value for the service is less than the threshold value. The threshold value also determines the load on each service.

Configuring Load Monitors

Nov 12, 2013

To configure a load monitor, first create the load monitor. For instructions on creating a monitor, see Creating Monitors. Next, select or create the metric table to define a set of metrics that determine the state of the server, and (if you create a metric table) bind each metric to the metric table.

To create a metric table by using the command line interface

At the command prompt, type the following commands:

- add lb metricTable <metricTableName>
- bind lb metricTable <metricTableName> <metric> <SNMPOID>

Example

```
add metricTable Table-Custom-1

bind metricTable Table-Custom-1 1.3.6.1.4.1.5951.4.1.1.41.1.5 11
```

To create a metric table and bind metrics to it by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Metric Tables.
2. In the details pane, click Add.
3. In the Create Metric Table dialog box, in the Metric Table Name text box, type the name of the metric table (for example, Table-Custom-1).
4. Click Create.
5. In the details pane, select the metric table that you just created (for example, Table-Custom-1), and then click Open.
6. In the Configure Metric Table dialog box, in the Metric and SNMP OID text boxes, type the metric and SNMP OID for the metric table (for example, 1.3.6.1.4.1.5951.4.1.1.41.1.5 and 11).
7. Click Add.
8. Click Close. The metric table you created appears in the Metric Tables pane.

Unbinding Metrics from a Metrics Table

Nov 12, 2013

You can unbind metrics from a metrics table if the metrics need to be changed, or if you want to remove the metrics table entirely.

To unbind metrics from a metric table by using the command line interface

At the command prompt, type:

unbind lb metricTable <metricTable> <metric>

Example

unbind metricTable Table-Custom-1 1.3.6.1.4.1.5951.4.1.1.41.1.5

To unbind metrics from a metric table by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Metric Tables.
2. In the details pane, select the metric table from which you want to unbind the metrics (for example, Table-Custom-1), click Open.
3. In the Configure Metric Table dialog box, in the Bound Metrics list box, select the metric that you want to unbind from the table (for example, 1.3.6.1.4.1.5951.4.1.1.41.1.5).
4. Click Remove, and then click OK.

You can view the detail of all configured metric tables, such as name and type, to determine whether the metric table is internal or created and configured.

Removing a Load Monitoring Metric Table

Nov 12, 2013

You can remove a metric table from the NetScaler configuration.

Note: Before you can remove a metric table, you must unbind all metrics from it.

To remove a metric table by using the command line interface

At the command prompt, type:

rm lb metricTable <metricTable>

Example

rm metricTable <Table-Custom-1>

To remove a metric table by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Metric Tables.
2. In the details pane, select the metric table that you want to remove (for example, Table-Custom-1), and click Remove.
3. In the Remove dialog box, and then click Yes.

You can unbind a metric from a metric table to remove that metric from consideration.

Viewing Metrics Tables

Nov 12, 2013

You can view a metrics table and the metrics bound to it.

To view the metric tables by using the command line interface

At the command prompt, type:

show lb metricTable <metricTable>

Example

show metricTable Table-Custom-1

To view the metric tables by using the configuration utility

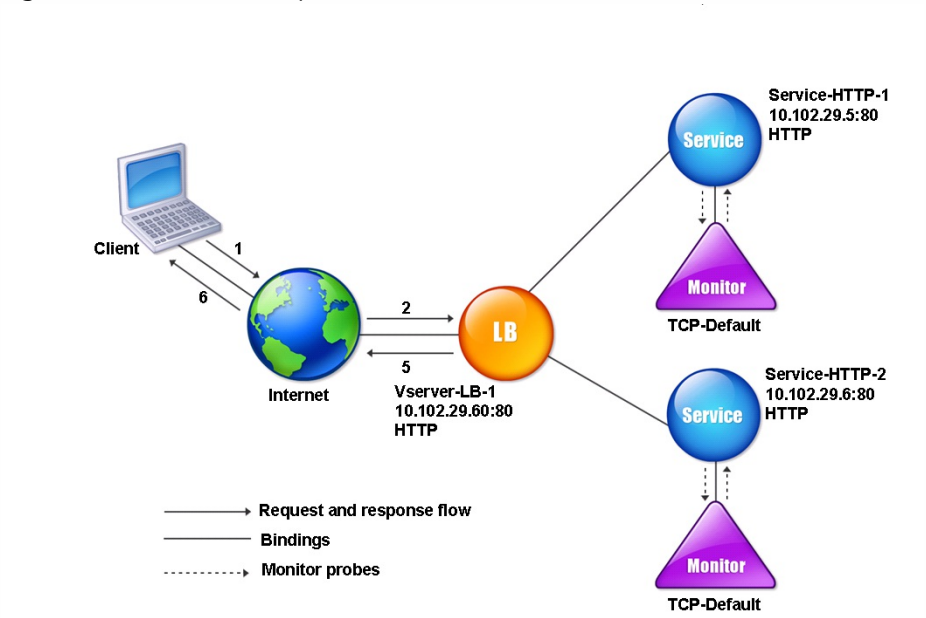
1. Navigate to Traffic Management > Load Balancing.
2. Click Metric Tables. The details of the available metric table appear on the Metric Tables pane.

Configuring Monitors in a Load Balancing Setup

Jun 09, 2015

To configure monitors on a Web site, you first decide whether to use a built-in monitor or create your own monitor. If you create a monitor, you can choose between creating a monitor based on a built-in monitor, or creating a custom monitor that uses a script that you write to monitor the service. For more information about creating custom monitors, see Custom Monitors. Once you have chosen or created a monitor, you then bind it to the appropriate service. The following conceptual diagram illustrates a basic load balancing setup with monitors.

Figure 1. How Monitors Operate



As shown above, each service has a monitor bound to it. The monitor probes the load balanced server via its service. As long as the load balanced server responds to the probes, the monitor marks it UP. If the load balanced server should fail to respond to the designated number of probes within the designated time period, the monitor marks it DOWN.

This section includes the following details:

- Creating Monitors
- Binding Monitors to Services
- Modifying Monitors
- Enabling and Disabling Monitors
- Unbinding Monitors
- Removing Monitors
- Viewing Monitors
- Closing Monitor Connections
- Ignoring the Upper Limit on Client Connections for Monitor Probes

Creating Monitors

Nov 12, 2013

The NetScaler appliance provides a set of built-in monitors. It also allows you to create custom monitors, either based on the built-in monitors or from scratch.

To create a monitor by using the command line interface

At the command prompt, type:

add lb monitor <monitorName> <monitorType> [<interval>]

Example

add lb mon monitor-HTTP-1 HTTP

add lb mon monitor-HTTP-2 TCP 2

To create a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. On the Monitors pane, click Add.
3. In the Create Monitor dialog box, in the Name and Interval text boxes type the name and interval value of the monitor.
4. In the Type list, select the type of the monitor.
5. In the list next to the Interval text box, select Seconds.
6. Click Create, and then click Close. The monitor that you created appears in the Monitors pane.

Binding Monitors to Services

Nov 12, 2013

After creating a monitor, you bind it to a service. You can bind one or multiple monitors to a service. If you bind one monitor to a service, that monitor determines whether the service is marked UP or DOWN. If you bind multiple monitors to a service, the NetScaler appliance checks all monitors bound to that service using a calculation that you control, and marks the service UP or DOWN depending on the results.

Note: The destination IP address of a monitor probe can be different than the server IP address and port.

To bind a monitor to a service by using the command line interface

At the command prompt, type:

bind lb monitor <monitorName> <ServiceName>

Example

bind mon monitor-HTTP-1 Service-HTTP-1

To bind a monitor to a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service for which you want to bind the monitor (for example, Service-HTTP-1), and then click Open.
3. On the Monitors tab, in the Available list box, select the monitor you want to bind the service (for example, monitor-HTTP-1), and then click Add.
4. In the Configured box, click OK.

Modifying Monitors

Nov 12, 2013

You can modify the settings for any monitor that you created.

Note: Two sets of parameters apply to monitors: those that apply to all monitors, regardless of type, and those that are specific to a monitor type. For information on parameters for a specific monitor type, see the description for that type of monitor.

To modify an existing monitor by using the command line interface

At the command prompt, type:

set lb monitor <monitorName> <type> -interval <interval> -resptimeout <resptimeout>

Example

```
set mon monitor-HTTP-1 HTTP -interval 50 milli
-resptimeout 20 milli
```

To modify an existing monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, select the monitor that you want to modify (for example, monitor-HTTP-1), and then click Open.
3. On the Standard Parameters tab, in the Interval and Response Time-out text boxes, type the interval and response timeout values (for example, 50 and 20).
4. In the list next to Interval text box, select the interval (for example, Milli Seconds).
5. In the list next to Response Time-out text box, select the interval (for example, Milli Seconds).
6. Click OK.

Enabling and Disabling Monitors

Nov 12, 2013

By default, monitors bound to services and service groups are enabled. When you enable a monitor, the monitor begins probing the services to which it is bound. If you disable a monitor bound to a service, the state the service is determined using the other monitors bound to the service. If the service is bound to only one monitor, and if you disable the monitor, the state of the service is determined using the default monitor.

To enable a monitor by using the command line interface

At the command prompt, type:

enable lb monitor <monitorName>

Example

enable lb mon monitor-HTTP-1

To enable a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. On the Monitors pane, select the monitor that you want to enable (for example, monitor-HTTP-1), and then click Enable.
3. In the Enable dialog box, click Yes.

To disable a monitor by using the command line interface

At the command prompt, type:

disable lb monitor <monitorName>

Example

disable lb mon monitor-HTTP-1

To disable a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. On the Monitors pane, select the monitor that you want to disable (for example, monitor-HTTP-1), and then click Disable.
3. In the Disable dialog box, click Yes.

Unbinding Monitors

Nov 12, 2013

You can unbind monitors from a service and service group. When you unbind a monitor from the service group, the monitors are unbound from the individual services that constitute the service group. When you unbind a monitor from a service or a service group, the monitor does not probe the service or the service group.

Note: When you unbind all user-configured monitors from a service or a service group, the default monitor is bound to the service and the service group. The default monitors then probes the service or the service groups.

To unbind a monitor from a service by using the command line interface

At the command prompt, type:

unbind lb monitor <monitorName>

Example

unbind mon monitor-HTTP-1 Service-HTTP-1

To unbind a monitor from a service by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service from that you want to unbind the monitor (for example, Service-HTTP-1), click Open.
3. In the Configure Service dialog box, under Configured, select the monitor that you want to unbind from the service (for example, monitor-HTTP-1), and then click Remove.
4. Click OK.

Removing Monitors

Nov 12, 2013

After you unbind a monitor that you created from its service, you can remove that monitor from the NetScaler configuration. (If a monitor is bound to a service, it cannot be removed.)

Note: When you remove monitors bound to a service, the default monitor is bound to the service. You cannot remove default monitors.

To remove a monitor by using the command line interface

At the command prompt, type:

rm lb monitor <monitorName> <type>

Example

rm lb monitor monitor-HTTP-1 HTTP

To remove a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. On the Monitors pane, select the monitor that you want to remove (for example, monitor-HTTP-1), and then click Remove.
3. In the Remove dialog box, click Yes.

Viewing Monitors

Nov 12, 2013

You can view the services and service groups that are bound to a monitor. You can verify the settings of a monitor to troubleshoot your NetScaler configuration. The following procedure describes the steps to view the bindings of a monitor to the services and service groups.

To view monitor bindings by using the command line interface

At the command prompt, type:

show lb monbindings <MonitorName>

Example

show lb monbindings monitor-HTTP-1

To view monitor bindings by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. On the Monitors pane, select the monitor for which you want to view the binding information (for example, monitor-HTTP-1), and then click Show Bindings. The binding information for the monitor that you selected appears in the Binding Info for Monitor: monitor-HTTP-1 dialog box.

To view monitors by using the command line interface

At the command prompt, type:

show lb monitor <monitorName>

Example

show lb mon monitor-HTTP-1

To view monitors by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors. The details of the available monitors appear in the Monitors pane.

Closing Monitor Connections

Aug 29, 2013

The NetScaler appliance sends probes to the services through the monitors bound to the services. By default, the monitor on the NetScaler and the physical server follow the complete handshake procedure even for monitor probes. However, this procedure adds overhead to the monitoring process and may not be always necessary.

For the TCP monitors, you can configure the NetScaler to close a monitor-probe connection after receiving SYN-ACK from the service. To do so, set the value of the monitorConnectionClose parameter to RESET. If you want the monitor-probe connection to go through the complete procedure, set the value to FIN.

Note: The monitorConnectionClose setting is applicable to all the monitors bound to all the services configured on the NetScaler appliance.

To configure monitor-connection closure by using the command line interface

At the command prompt, type:

set lb parameter -monitorConnectionClose <monitor_conn_close_option>

Example

set lb parameter -monitorConnectionClose RESET

To configure monitor-connection closure by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. Under Settings, click Configure Load Balancing Parameters.
3. In the Configure Load Balancing Parameters dialog box, for Connection Close for Monitor, select FIN or RESET.
4. Click OK.

Ignoring the Upper Limit on Client Connections for Monitor Probes

Aug 29, 2013

Depending on considerations such as the capacity of a physical server, you can specify a limit on the maximum number of client connections made to any service. If you have set such a limit on a service, the NetScaler appliance stops sending requests to the service when the threshold is reached and resumes sending connections to the service after the number of existing connections falls to within the limits. You can configure the NetScaler to skip this check when it sends monitor-probe connections to a service.

Note: You cannot skip the maximum-client-connections check for an individual service. If you specify this option, it applies to all the monitors bound to all the services configured on the NetScaler appliance.

To set the Skip MaxClients for Monitor Connections option by using the command line interface

At the command prompt, type:

set lb parameter -monitorSkipMaxClient (ENABLED | DISABLED)

Example

```
set lb parameter -monitorSkipMaxClient enabled
```

To set the Skip MaxClients for Monitor Connections option by using the configuration utility

1. Navigate to Traffic Management > Load Balancing.
2. Under Settings, click Configure Load Balancing Parameters.
3. To ignore the upper limit on client connections for monitor probes, in the Configure Load Balancing Parameters dialog box, select the Skip MaxClients for Monitor Connections checkbox.
4. Click OK.

Managing a Large Scale Deployment

Mar 19, 2012

The NetScaler appliance contains several features that are helpful when you are configuring a large load balancing deployment. Instead of configuring virtual servers and services individually, you can create groups of virtual servers and services. You can also create a range of virtual servers and services, and you can translate or mask virtual server and service IP addresses.

You can set persistence for a group of virtual servers. You can bind monitors to a group of services. Creating a range of virtual servers and services of identical type allows you to set up and configure those servers in a single procedure, which significantly shortens the time required to configure those virtual servers and services.

By translating or masking IP addresses, you can take down virtual servers and services, and make changes to your infrastructure, without extensive reconfiguration of your service and virtual server definitions.

Ranges of Virtual Servers and Services

Mar 19, 2012

When you configure load balancing, you can create ranges of virtual servers and services, eliminating the need to configure virtual servers and services individually. For example, you can use a single procedure to create three virtual servers with three corresponding IP addresses. When more than one argument uses a range, all of the ranges must be of the same size.

The following are the types of ranges you can specify when adding services and virtual servers to your configuration:

- **Numeric ranges.** Instead of typing a single number, you can specify a range of consecutive numbers.

  For example, you can create a range of virtual servers by specifying a starting IP address, such as 10.102.29.30, and then typing a value for the last byte that indicates the range, such as 34. In this example, five virtual servers will be created with IP addresses that range between 10.102.29.30 and 10.102.29.34.

  Note: The IP addresses of the virtual servers and services must be consecutive.
- **Alphabetic ranges.** Instead of typing a literal letter, you can substitute a range for any single letter, for example, [C-G]. This results in all letters in the range being included, in this case C, D, E, F, and G.

  For example, if you have three virtual servers named Vserver-x, Vserver-y, and Vserver-z, instead of configuring them separately, you can type vserver [x-z] to configure them all.

Creating a Range of Virtual Servers

Updated: 2013-11-12

You create a range of virtual servers as described below.

To create range of virtual servers by using the command line interface

At the command prompt, type one of the following commands:

- add lb vserver <name>@ <protocol> -range <rangeValue> <IPAddress> [<port>]
- add lb vserver <name>@**[**<rangeValue>**]**> <protocol> <IPAddress[<rangeValue>]> [<port>]

Example

```
add lb vserver Vserver-LB-2 http -range 6 10.102.29.30 80
```

OR

```
> add lb vserver vserver[P-R] http 10.102.29.[26-28] 80
vserver "vserverP" added
vserver "vserverQ" added
vserver "vserverR" added
 Done
```

To create range of virtual servers by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add Range.

3. In the Create Virtual Server (Load Balancing) - Range dialog box, in the Name Prefix, IP Address Range, and Port text boxes, type the virtual server name, IP address with which to begin the range, and port.
4. Select the Network VServer check box, and in Range, type the last value of the virtual server range.
5. In the Protocol drop-down list box, select the protocol type.
6. Click Create, and then click Close. The range of virtual servers you created appears in the Load Balancing Virtual Servers pane.

Creating a Range of Services

Updated: 2013-11-12

You create a range of services as described below. If you specify a range for the service name, specify a range for the IP address too.

To create range of services by using the command line interface

At the command prompt, type the command:

add service <name>@ <IP>@ <protocol> <port>
Example

```
> add service serv[1-3] 10.102.29.[102-104] http 80
service "serv1" added
service "serv2" added
service "serv3" added
 Done
```

To create range of services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add Range.
3. In the Create Service (Range) dialog box, in the IP Address Range and Port text boxes, type the start value of the IP address range and the port.
4. In the text box next to the IP Address Range text box, type the last value of the last service (for example, 104).
5. In the Protocol drop-down list box, select the protocol type.
6. Click Create, and then click Close. The range of services you created appears in the Services pane.

Configuring Service Groups

Mar 19, 2012

Configuring a service group enables you to manage a group of services as easily as a single service. For example, if you enable or disable any option, such as compression, health monitoring or graceful shutdown, for a service group, the option gets enabled for all the members of the service group.

After creating a service group, you can bind it to a virtual server, and you can add services to the group. You can also bind monitors to service groups.

The members of a service group can be identified by IP address or server name.

Using domain-name based service (DBS) group members is advantageous because you need not reconfigure the member on the NetScaler appliance if the IP address of the member changes. The appliance automatically senses such changes through the configured name server. This feature is particularly useful in cloud scenarios, where the service provider can change a physical server or change the IP address for a service. If you specify a DBS group member, the NetScaler learns the IP address dynamically.

You can bind both IP-based and DBS members to the same service group.

Note: If you use DBS service group members, make sure that either a name server is specified or a DNS server is configured on the NetScaler. A domain name will be resolved into an IP address only if the corresponding address record is present on the NetScaler or the name server.

Creating Service Groups

Updated: 2013-09-04

You can configure up to 8192 service groups on the NetScaler appliance.

To create a service group by using the command line

At the command prompt, type:

add servicegroup <ServiceGroupName> <Protocol>

Example

```
add servicegroup Service-Group-1 HTTP
```

To create a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, click Add.
3. In the Create Service Group dialog box, in the Service Group Name text box, type name of the service group.
4. In the Protocol list, select the protocol type.
5. Click Create, and then click Close. The service group you created appears in the Service Groups pane.

Binding a Service Group to a Virtual Server

Updated: 2013-11-12

When you bind a service group to a virtual server, the member services are bound to the virtual server.

To bind a service group to a virtual server by using the command line interface

At the command prompt, type:

bind lb vserver <name>@ <serviceGroupName>
Example

```
bind lb vserver Vserver-LB-1 Service-Group-1
```

To bind a service group to a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind the service group, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Services Groups tab.
4. In the Active column, select check box next to the service group that you want to bind to the virtual server (for example, Service-Group-1), and then click OK.

Binding a Member to a Service Group

Updated: 2013-11-12

Adding services to a service group enables the service group to manage the servers. You can add the servers to a service group by specifying the IP addresses or the names of the servers.

To add members to a service group by using the command line interface

To configure a service group, at the command prompt, type:

bind servicegroup <serviceGroupName> (<IP>@ | <serverName>) <port>
Examples

```
bind servicegroup Service-Group-1 10.102.29.30 80

bind servicegroup Service-Group-2 1000:0000:0000:0000:0005:0600:700a:888b 80

bind servicegroup CitrixEdu s1.citrite.net
```

To add members to a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group to which you want to bind members, and then click Open.
3. In the Configure Service Group dialog box, under Specify Member(s), do one of the following:
   - To add a new IP based service group member, select IP Based.
   - To add a server-name based service group member, select Server Based.

   If you want to add a domain-name based service group member, select **Server Based**. With this option, you can add any server that has been assigned a name, regardless of whether the name is an IP address or a user-assigned name.
4. If adding a new IP based member, in the IP Address text box, type the IP address. If the IP address uses IPv6 format, select the IPv6 check box and then enter the address in the IP Address text box.

   Note: You can add a range of IP addresses. The IP addresses in the range must be consecutive. Specify the range by entering the starting IP address in the IP Address text box (for example, 10.102.29.30). Specify the end byte of the IP address range in the text box under Range (for example, 35). In the Port text box type the port (for example, 80), and

then click Add.

5. Click OK.

Binding a Monitor to a Service Group

Updated: 2013-12-10

When you create a service group, the default monitor of the type appropriate for the group is automatically bound to it. Monitors periodically probe the servers in the service group to which they are bound and update the state of the service groups.

You can bind a different monitor of your own choice to the service group.

To bind a monitor to a service group by using the command line interface

At the command prompt, type:

bind serviceGroup <serviceGroupName> -monitorName <string> -monState (ENABLED | DISABLED)

Example

bind serviceGroup Service-Group-1 -monitorName monitor-HTTP-1

To a bind monitor to a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group for which you want to bind monitors, and then click Open.
3. On the Monitors tab, under Available, select a monitor name.
4. Click Add, and then click OK.

Managing Service Groups

Jun 04, 2015

You can change the settings of the services in a service group, and you can perform tasks such as enabling, disabling, and removing service groups. You can also unbind members from a service group.

To manage service groups, see the following sections:

- Modifying a Service Group
- Removing a Service Group
- Unbinding a Member from a Service Group
- Unbinding a Service Group from a Virtual Server
- Unbinding Monitors from Service Groups
- Enabling or Disabling a Service Group
- Viewing the Properties of a Service Group
- Viewing Service Group Statistics
- Load Balancing Virtual Servers Bound to a Service Group

Modifying a Service Group

Updated: 2013-11-12

You can modify attributes of service group members. You can set several attributes of the service group, such as maximum client, SureConnect, and compression. The attributes are set on the individual servers in the service group. You cannot set parameters on the service group such as transport information (IP address and port), weight, and server ID.

Note: A parameter you set for a service group is applied to the member servers in the group, not to individual services.

To modify a service group by using the command line interface

At the command prompt, type the following command with one or more of the optional parameters:

set servicegroup <serviceGroupName> [-type <type>] [-maxClient <maxClient>] [-maxReq <maxReq>] [-cacheable (YES | NO)] [-cip (ENABLED | DISABLED)] [-cipHeader <cipHeader>] [-usip (YES | NO)] [-sc (ON | OFF)] [-sp (ON | OFF)] [-cltTimeout <cltTimeout>] [-svrTimeout <svrTimeout>] [-cka (YES | NO)] [-TCPB (YES | NO)] [-CMP (**YES** | **NO**)] [-maxBandwidth <maxBandwidth>] [-maxThreshold <maxThreshold>] [-state (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)

Example

```
set servicegroup Service-Group-1 -type TRANSPARENT

set servicegroup Service-Group-1 -maxClient 4096

set servicegroup Service-Group-1 -maxReq 16384

set servicegroup Service-Group-1 -cacheable YES
```

To modify a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group that you want to modify (for example, Service-Group-1), and then click Open.

3. Change any of the parameters and then click OK.

Removing a Service Group

Updated: 2013-09-04

When you remove a service group, the servers bound to the group retain their individual settings and continue to exist on the NetScaler.

To remove a service group by using the command line interface

At the command prompt, type:

rm servicegroup <ServiceGroupName>
Example

rm servicegroup Service-Group-1

To remove a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group that you want to remove (for example, Service-Group-1), and then click Remove.
3. In the Remove dialog box, click Yes.

Unbinding a Member from a Service Group

Updated: 2013-11-12

When you unbind a member from the service group, the attributes set on the service group will no longer apply to the member that you unbound. The member services retains its individual settings, however, and continues to exist on the NetScaler.

To unbind members from a service group by using the command line interface

At the command prompt, type:

unbind servicegroup <serviceGroupName> <IP>@ [<port>]
Example

unbind servicegroup Service-Group-1 10.102.29.30 80

To unbind members from a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group from which you want to unbind members, and then click Open.
3. In the Configure Service Group dialog box, in the Configured Members list box, select a service.
4. Click Remove, and then click OK.

Unbinding a Service Group from a Virtual Server

Updated: 2013-11-12

When you unbind a service group from a virtual server, the member services are unbound from the virtual server and

continue to exist on the NetScaler appliance.

To unbind a service group from a virtual server by using the command line interface

At the command prompt, type:

unbind lb vserver <name>@ <ServiceGroupName>
Example

```
unbind lb vserver Vserver-LB-1 Service-Group-1
```

To unbind a service group from a virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server from which you want to unbind the service group, and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Services Group tab.
4. Clear the Active check box next to the service group that you want to unbind from the virtual server (for example, Service-Group-1).
5. Click OK.

Unbinding Monitors from Service Groups

Updated: 2013-12-10

When you unbind a monitor from a service group, the monitor that you unbound no longer monitors the individual services that constitute the group.

To unbind a monitor from a service group using the command line interface

At the command prompt, type:

unbind serviceGroup <serviceGroupName> -monitorName <string>
Example

```
unbind serviceGroup Service-Group-1 -monitorName monitor-HTTP-1
```

To unbind a monitor from a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group from which you want to unbind the monitor, click Open.
3. In the Configure Service Group dialog box, click the Monitors tab.
4. Under Configured, select the monitor that you want to unbind from the service group, and then click Remove.
5. Click OK.

Enabling or Disabling a Service Group

Updated: 2013-09-04

When you enable a service group and the servers, the services belonging to the service group are enabled. Similarly, when a service belonging to a service group is enabled, the service group and the service are enabled. By default, service groups are enabled.

After disabling an enabled service, you can view the service using the configuration utility or the command line to see the amount of time that remains before the service goes DOWN.

To disable a service group by using the command line interface

At the command prompt, type:

disable servicegroup <ServiceGroupName>
Example

```
disable servicegroup Service-Group-1
```

To disable a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the Service Groups pane, select the service group that you want to disable, and then click Disable.
3. In the Wait Time dialog box type the wait time value.
4. Click Enter.

To enable a service group by using the command line interface

At the command prompt, type:

enable servicegroup <ServiceGroupName>
Example

```
enable servicegroup Service-Group-1
```

To enable a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the Service Groups pane, select the service group that you want to enable (for example, Service-Group-1), and then click Enable.
3. In the Enable dialog box, click Yes.

Viewing the Properties of a Service Group

Updated: 2013-09-04

You can view the following settings of the configured service groups: name, IP address, state, protocol, maximum client connections, maximum requests per connection, maximum bandwidth, and monitor threshold. Viewing the details of the configuration can be helpful for troubleshooting your configuration.

To view the properties of a service group by using the command line interface

At the command prompt, type one of the following commands to display the group properties or the properties and the group members:

- show servicegroup <ServiceGroupName>
- show servicegroup <ServiceGroupName> -includemembers

Example

```
show servicegroup Service-Group-1
```

To view the properties of a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, click the name of the service group whose properties you want to view, and then click Open.

Viewing Service Group Statistics

Updated: 2013-09-04

You can view service-group statistical data, such as rate of requests, responses, request bytes, and response bytes. The NetScaler appliance uses the statistics of a service group, such as these, to balance the load on the services.

To view the statistics of a service group by using the command line interface

At the command prompt, type:

stat servicegroup <ServiceGroupName>
Example

```
stat servicegroup Service-Group-1
```

To view the statistics of a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, select the service group for which statistics you want to view (for example, Service-Group-1), and then click Statistics. The statistics of the service group you selected appear in a new window.

Load Balancing Virtual Servers Bound to a Service Group

Updated: 2013-09-04

In large-scale deployments, the same service group can be bound to multiple load balancing virtual servers. In such a case, instead of viewing each virtual server to see the service group it is bound to, you can view a list of all the load balancing virtual servers bound to a service group. You can view the following details of each virtual server:

- Name
- State
- IP address
- Port

To display the virtual servers bound to a service group by using the command line interface

At the command prompt, type the following command to display the virtual servers bound to a service group:

show servicegroupbindings <serviceGroupName>
Example

```
> show servicegroupbindings SVCGRPDTLS
SVCGRPDTLS - State :ENABLED
1) Test-pers (10.10.10.3:80) - State : DOWN
```



```
2) BRVSERV (10.10.1.1:80) - State : DOWN
3) OneMore (10.102.29.136:80) - State : DOWN
4) LBVIP1 (10.102.29.66:80) - State : UP
Done
>
```

To display the virtual servers bound to a service group by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. Select a service group, and then click Show Bindings. The Binding details for Service Group box then displays all the load balancing virtual servers bound to the selected service group.

Configuring Automatic Domain Based Service Group Scaling

Sep 04, 2013

A domain based service group consists of members whose IP addresses are obtained by resolving the domain names of servers that are bound to the service group. The domain names are resolved by a name server whose details you configure on the appliance. A domain based service group can also include IP-address based members.

The process of name resolution for a domain based server might return more than one IP address. The number of IP addresses in the DNS response is determined by the number of address (A) records configured for the domain name, on the name server. Even if the name resolution process returns multiple IP addresses, only one IP address is bound to the service group. To scale up or scale down a service group, you need to manually bind and unbind additional domain based servers to and from the service group, respectively.

However, you can configure a domain based service group to scale automatically on the basis of the complete set of IP addresses returned by a DNS name server for a domain based server. To configure automatic scaling, when binding a domain based server to a service group, enable the automatic scaling option. Following are the steps for configuring a domain based service group that scales automatically:

- Add a name server for resolving domain names. For more information about configuring a name server on the appliance, see Adding a Name Server.
- Add a domain based server. For information about adding a domain based server, see Adding a Server.
- Add a service group and associate the domain based server to the service group, with the autoscale option set to DNS. For information about adding a service group, see Configuring Service Groups.

When a domain based server is bound to a service group and the automatic scaling option is set on the binding, a UDP monitor and a TCP monitor are automatically created and bound to the domain based server. The two monitors function as resolvers. The TCP monitor is disabled by default, and the appliance uses the UDP monitor to send DNS queries to the name server to resolve the domain name. If the DNS response is truncated (has the TC flag set to 1), the appliance falls back to TCP and uses the TCP monitor to send the DNS queries over TCP. Thereafter, the appliance continues to use only the TCP monitor.

The DNS response from the name server might contain multiple IP addresses for the domain name. With the automatic scaling option set, the appliance polls each of the IP addresses by using the default monitor, and then includes in the service group only those IP addresses that are up and available. After the IP address records expire, as defined by their time-to-live (TTL) values, the UDP monitor (or the TCP monitor, if the appliance has fallen back to using the TCP monitor) queries the name server for domain resolution and includes any new IP addresses in the service group. If an IP address that is part of the service group is not present in the DNS response, the appliance removes that address from the service group after gracefully closing existing connections to the group member, a process during which it does not allow any new connections to be established with the member. If a domain name that resolved successfully in the past results in an NXDOMAIN response, all the service group members associated with that domain are removed.

Static (IP-address based) members and dynamically scaling domain based members can coexist in a service group. You can also bind members with different domain names to a service group with the automatic scaling option set. However, each domain name associated with a service group must be unique within the service group. You must enable the automatic scaling option for each domain based server that you want to use for automatic service group scaling. If an IP address is common to one or more domains, the IP address is added to the service group only once.

To configure a service group to scale automatically by using the command line interface

At the command prompt, type the following commands to configure the service group and verify the configuration:

- bind serviceGroup <serviceGroupName> <serverName> <port> -autoScale (**YES** | **NO**)
- show serviceGroup <serviceGroupName>

Example

In the following example, server1 is a domain based server. The DNS response contains multiple IP addresses. Five addresses are available and are added to the service group.

```
> bind serviceGroup servGroup server1 80 -autoScale YES
 Done
> sh servicegroup servGroup
      servGroup - HTTP
       State: ENABLED  Monitor Threshold : 0
           . . .
           . . .
       1)  192.0.2.31:80   State: UP      Server Name: server1 (Auto scale)   Server ID: None Weight: 1

            Monitor Name: tcp-default      State: UP
            Probes: 2      Failed [Total: 0 Current: 0]
            Last response: Success - TCP syn+ack received.

       2)  192.0.2.32:80   State: UP      Server Name: server1 (Auto scale)   Server ID: None Weight: 1

            Monitor Name: tcp-default      State: UP
            Probes: 2      Failed [Total: 0 Current: 0]
            Last response: Success - TCP syn+ack received.

       3)  192.0.2.36:80   State: UP      Server Name: server1 (Auto scale)   Server ID: None Weight: 1

            Monitor Name: tcp-default      State: UP
            Probes: 2      Failed [Total: 0 Current: 0]
            Last response: Success - TCP syn+ack received.

       4)  192.0.2.55:80   State: UP      Server Name: server1 (Auto scale)   Server ID: None Weight: 1

            Monitor Name: tcp-default      State: UP
            Probes: 2      Failed [Total: 0 Current: 0]
            Last response: Success - TCP syn+ack received.

       5)  192.0.2.80:80   State: UP      Server Name: server1 (Auto scale)   Server ID: None Weight: 1

            Monitor Name: tcp-default      State: UP
            Probes: 2      Failed [Total: 0 Current: 0]
            Last response: Success - TCP syn+ack received.
 Done
```

To configure a service group to scale automatically by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Service Groups.
2. In the details pane, do one of the following:
   - To create a domain based service, click Add.
   - To bind a domain based server to an existing service group, click the name of the service group, and then click Open.
3. In the Create Service Group or Configure Service Group dialog box, set the following parameters:
   - Server Based (Specifies that you want to bind a domain based server to the service group. A list that contains all the IP addresses and servers configured on the NetScaler appliance appears. Select the domain based server that you want.)
   - Port
   - Auto Scale
     After you set the parameters, to bind the server to the service group, click Add.
4. Click Create or OK, and then click Close.

Translating the IP Address of a Domain-Based Server

Nov 12, 2013

To simplify maintenance on the NetScaler appliance and on the domain-based servers that are connected to it, you can configure IP address masks and translation IP addresses. These functions work together to parse incoming DNS packets and substitute a new IP address for a DNS-resolved IP address.

When configured for a domain-based server, IP address translation enables the appliance to locate an alternate server IP address in the event that you take the server down for maintenance or if you make any other infrastructure changes that affect the server.

When configuring the mask, you must use standard IP mask values (a power of two, minus one) and zeros, for example, 255.255.0.0. Non-zero values are only permitted in the starting octets.

When you configure a translation IP for a server, you create a 1:1 correspondence between a server IP address and an alternate server that shares leading or trailing octets in its IP address. The mask blocks particular octets in the original server's IP address. The DNS-resolved IP address is transformed to a new IP address by applying the translation IP address and the translation mask.

For example, you can configure a translation IP address of 10.20.0.0 and a translation mask of 255.255.0.0. If a DNS-resolved IP address for a server is 40.50.27.3, this address is transformed to 10.20.27.3. In this case, the translation IP address supplies the first two octets of the new address, and the mask passes through the last two octets from the original IP address. The reference to the original IP address, as resolved by DNS, is lost. Monitors for all services to which the server is bound also report on the transformed IP address.

When configuring a translation IP address for a domain-based server, you specify a mask and an IP address to which the DNS-resolved IP address is to be translated.

Note: Translation of the IP address is only possible for domain-based servers. You cannot use this feature for IP-based servers. The address pattern can be based on IPv4 addresses only.

To configure a translation IP address for a server by using the command line interface

At the command prompt, type:

add server <name>@ <serverDomainName> -translationIp <translationIPAddress> -translationMask <netMask> -state <ENABLED|DISABLED>

Example

```
add server myMaskedServer www.example.com -translationIp 10.10.10.10 -translationMask
255.255.0.0 -state ENABLED
```

To configure a translation IP address for a server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Servers.
2. In the details pane, click Add.
3. In the Create Server dialog box, in the Server Name field, enter a name.
4. In the IP Address / Domain Name field, enter the server's domain name.
   Note: Do not enter an IP address if you are entering a mask.
5. In the Translation IP Address field, enter the IP address of a server on the same subnet.
6. In the Translation Mask field, enter a valid mask (for example, 255.255.0.0).

7. Click Create.

Masking a Virtual Server IP Address

Nov 12, 2013

You can configure a mask and a pattern instead of a fixed IP address for a virtual server. This enables traffic that is directed to any of the IP addresses that match the mask and pattern to be rerouted to a particular virtual server. For example, you can configure a mask that allows the first three octets of an IP address to be variable, so that traffic to 111.11.11.198, 22.22.22.198, and 33.33.33.198 is all sent to the same virtual server.

By configuring a mask for a virtual server IP address, you can avoid reconfiguration of your virtual servers due to a change in routing or another infrastructure change. The mask allows the traffic to continue to flow without extensive reconfiguration of your virtual servers.

The mask for a virtual server IP address works somewhat differently from the IP pattern definition for a server described in Translating the IP Address of a Domain-Based Server. For a virtual server IP address mask, a non-zero mask is interpreted as an octet that is considered. For a service, the non-zero value is blocked.

Additionally, for a virtual server IP address mask, either leading or trailing values can be considered. If the virtual server IP address mask considers values from the left of the IP address, this is known as a forward mask. If the mask considers the values to the right side of the address, this is known as a reverse mask.

Note: The NetScaler appliance evaluates all forward mask virtual servers before evaluating reverse mask virtual servers. When masking a virtual server IP address, you also need to create an IP address pattern for matching incoming traffic with the correct virtual server. When the appliance receives an incoming IP packet, it matches the destination IP address in the packet with the bits that are considered in the IP address pattern, and after it finds a match, it applies the IP address mask to construct the final destination IP address.

Consider the following example:

- Destination IP address in the incoming packet: 10.102.27.189
- IP address pattern: 10.102.0.0
- IP mask: 255.255.0.0
- Constructed (final) destination IP address: 10.102.27.189.

In this case, the first 16 bits in the original destination IP address match the IP address pattern for this virtual server, so this incoming packet is routed to this virtual server.

If a destination IP address matches the IP patterns for more than one virtual server, the longest match takes precedence. Consider the following example:

- Virtual Server 1: IP pattern 10.10.0.0, IP mask 255.255.0.0
- Virtual Server 2: IP pattern 10.10.10.0, IP mask 255.255.255.0
- Destination IP address in the packet: 10.10.10.45.
- Selected virtual server: Virtual Server 2.

The pattern associated with Virtual Server 2 matches more bits than that associated with Virtual Server 1, so IPs that match it will be sent to Virtual Server 2.

Note: Ports are also considered if a tie-breaker is required.

To configure a virtual server IP address mask by using the command line interface

At the command prompt, type:

add lb vserver <name>@ http -ipPattern <ipAddressPattern> -ipMask <ipMask> <listenPort>

Example

Pattern matching based on prefix octets:

```
add lb vserver myLBVserver http -ippattern 10.102.0.0 -ipmask 255.255.0.0 80
```

Pattern matching based on trailing octets:

```
add lb vserver myLBVserver1 http -ippattern 0.0.22.74 -ipmask 0.0.255.255 80
```

Modify a pattern-based virtual server:

```
set lb vserver myLBVserver1 -ippattern 0.0.22.74 -ipmask 0.0.255.255
```

To configure a virtual server IP address mask by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server dialog box, in the Name field, enter a name.
4. In the Protocol field, select the protocol.
5. In the Port field, enter the listen port.
6. In the IP Pattern field, enter a pattern for an IP address. The fixed part of the pattern must be entered in contiguous octets. Enter zeros for the pattern values that can vary in the IP address.
7. In the IP Mask field, enter a standard network mask. Use non-zero mask values for the portion of the IP address that constitutes the fixed part of the pattern.

Configuring Load Balancing for Commonly Used Protocols

Jun 05, 2015

In addition to Web sites and Web-based applications, other types of network-deployed applications that use other common protocols often receive large amounts of traffic and therefore benefit from load balancing. Several of these protocols require specific configurations for load balancing to work properly. Among them are FTP, DNS, SIP, and RTSP.

If you configure your NetScaler appliance to use domain names for your servers rather than IPs, you may also need to set up IP translation and masking for those servers.

To configure load balancing for commonly used protocols, see the following sections:

- Load Balancing for a Group of FTP Servers
- Load Balancing DNS Servers
- Load Balancing Domain-Name Based Services
- Load Balancing a Group of SIP Servers
- Load Balancing RTSP Servers
- Load Balancing of Remote Desktop Protocol (RDP) Servers

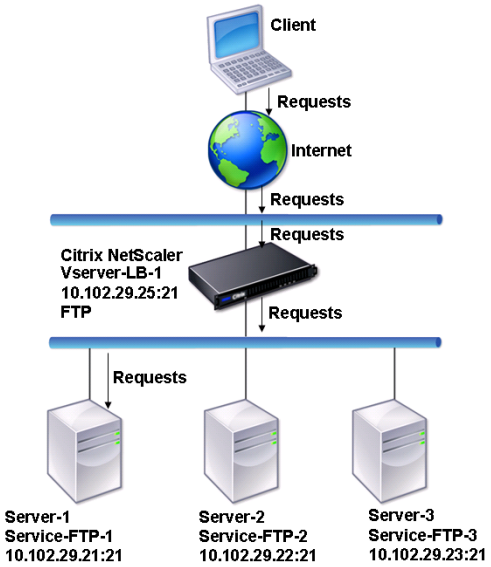
Load Balancing for a Group of FTP Servers

Feb 11, 2015

The NetScaler appliance can be used to load balance FTP servers. FTP requires that the user initiate two connections on two different ports to the same server: the control connection, through which the client sends commands to the server, and the data connection, through which the server sends data to the client. When the client initiates an FTP session by opening a control connection to the FTP server, the appliance uses the configured load balancing method to select an FTP service, and forwards the control connection to it. The load balanced FTP server then opens a data connection to the client for information exchange.

The following diagram describes the topology of a load balancing configuration for a group of FTP servers.

Figure 1. Basic Load Balancing Topology for FTP Servers



In the diagram, the services Service-FTP-1, Service-FTP-2, and Service-FTP-3 are bound to the virtual server Vserver-LB-1. Vserver-LB-1 forwards the client's connection request to one of the services using the least connection load balancing method. Subsequent requests are forwarded to the service that the appliance initially selected for load balancing.

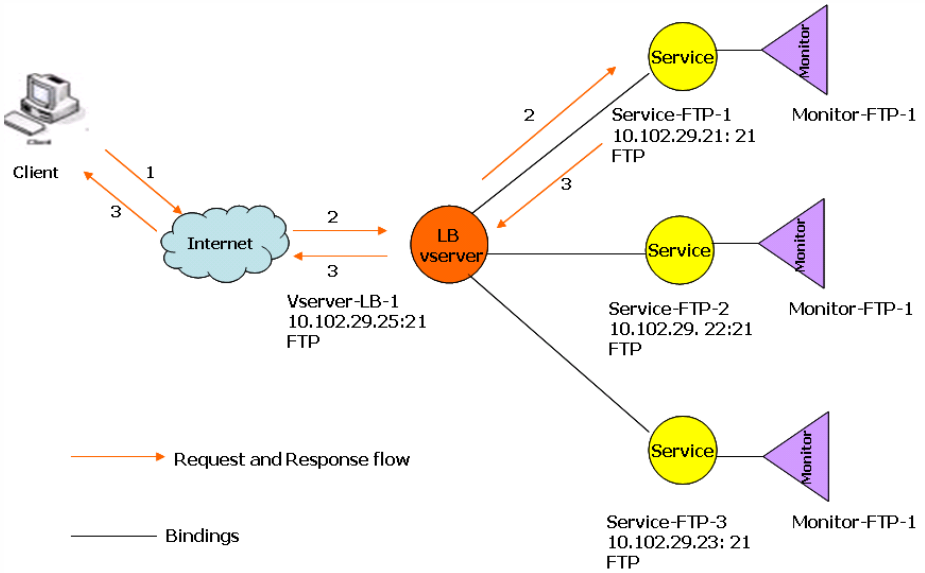
The following table lists the names and values of the basic entities configured on the appliance.

Entity type	Name	IP address	Port	Protocol
Vserver	Vserver-LB-1	10.102.29.25	21	FTP
Services	Service-FTP-1	10.102.29.21	21	FTP
	Service-FTP-2	10.102.29.22	21	FTP
	Service-FTP-3	10.102.29.23	21	FTP

Entity type	Name	IP address	Port	Protocol
Monitors	FTP	None	None	None

The following diagram shows the load balancing entities, and the values of the parameters that need to be configured on the appliance.

Figure 2. Load Balancing FTP Servers Entity Model



The appliance can also provide a passive FTP option to access FTP servers from outside of a firewall. When a client uses the passive FTP option and initiates a control connection to the FTP server, the FTP server also initiates a control connection to the client. It then initiates a data connection to transfer a file through the firewall.

To create services and virtual servers of type FTP, see Setting Up Basic Load Balancing. Name the entities and set the parameters to the values described in the columns of the previous table. When you configure a basic load balancing setup, a default monitor is bound to the services.

Next, bind the FTP monitor to the services by following the procedure described in the section Binding Monitors to Services.

To create FTP monitors by using the command line interface

At the command prompt, type:

add lb monitor <MonitorName> FTP -interval <Interval> -userName <UserName> -password <Password>

Example

add lb monitor monitor-FTP-1 FTP -interval 360 -userName User -password User

To create FTP monitors by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. On the Standards Parameters tab, in the Name and Interval text boxes, type the monitor name and interval.
4. In the Type list, select FTP.
5. On the Special Parameters tab, in the User Name and Password text boxes, type User.
6. Click Create, and then click Close. The monitor that you created appears in the Monitors pane.

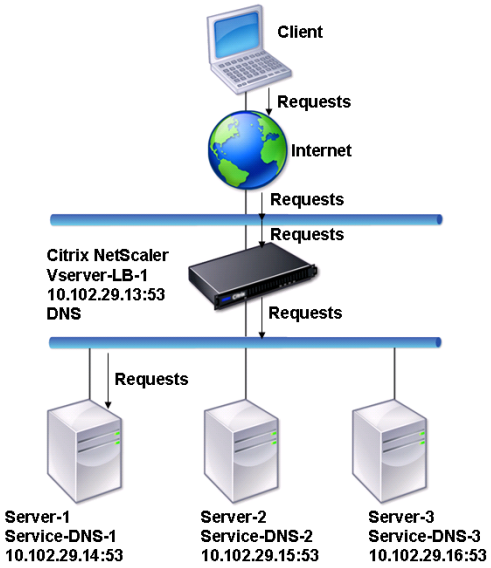
Load Balancing DNS Servers

Sep 03, 2013

When you request DNS resolution of a domain name, the NetScaler appliance uses the configured load balancing method to select a DNS service. The DNS server to which the service is bound then resolves the domain name and returns the IP address as the response. The appliance can also cache DNS responses and use the cached information to respond to future requests for resolution of the same domain name. Load balancing DNS servers improves DNS response times.

The following diagram describes the topology of a load balancing configuration that load balances a group of DNS services.

Figure 1. Basic Load Balancing Topology for DNS Servers

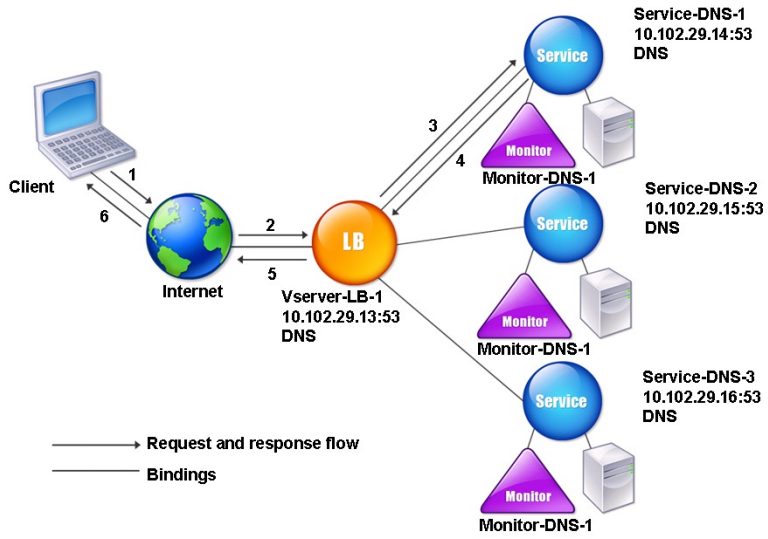


In the diagram, the services Service-DNS-1, Service-DNS-2, and Service-DNS-3 are bound to the virtual server Vserver-LB-1. The virtual server Vserver-LB-1 forwards client requests to a service using the least connection load balancing method. The following table lists the names and values of the basic entities configured on the appliance.

Entity type	Name	IP address	Port	Protocol
Virtual Server	Vserver-LB-1	10.102.29.13	53	DNS
Services	Service-DNS-1	10.102.29.14	53	DNS
	Service-DNS-2	10.102.29.15	53	DNS
	Service-DNS-3	10.102.29.16	53	DNS
Monitors	monitor-DNS-1	None	None	None

The following diagram shows the load balancing entities and the values of the parameters that need to be configured on the appliance.

Figure 2. Load Balancing DNS Servers Entity Model



To configure a basic DNS load balancing setup, see Setting Up Basic Load Balancing. Follow the procedures to create services and virtual servers of type DNS, naming the entities and setting the parameters using the values described in the previous table. When you configure a basic load balancing setup, the default ping monitor is bound to the services. For instructions on binding a DNS monitor to DNS services, you can also see Binding Monitors to Services.

The following procedure describes the steps to create a monitor that maps a domain name to the IP address based on a query.

To configure DNS monitors by using the command line interface

At the command prompt, type:

add lb monitor <monitorName> DNS -query <domainName> -queryType <**Address** | **ZONE**> -IPAddress <ipAddress>
Example

```
add lb monitor monitor-DNS-1 DNS -query www.citrix.com -queryType Address -IPAddress 10.102.29.66

add lb monitor monitor-DNS-2 DNS -query www.citrix2.com -queryType Address -IPAddress
1000:0000:0000:0000:0005:0600:700a::888b-888d
```

To configure DNS monitors by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, in the Name and Interval text boxes, type a monitor name and a monitoring interval.
4. Select the unit of time for the interval in the drop-down menu.
5. In the Type list, select DNS.
6. Click the Special Parameters tab, in the Query text box type the domain name query to send to the DNS service (for example, www.mycompany.com), and in the Query Type list box, select ADDRESS or ZONE.

7. In the text box below the Query Type list box, type an IP address that is to be checked against the response to the DNS monitoring query (for example, 10.102.29.66), and click Add.
   Note: If you want to enter an IPv6 address, select the IPv6 check box before entering the address.
8. Click Create, and then click Close. The monitor that you created appears in the Monitors pane.

Load Balancing Domain-Name Based Services

Nov 12, 2013

When you create a service for load balancing, you can provide an IP address. Alternatively, you can create a server using a domain name. The server name (domain name) can be resolved using an IPv4 or IPv6 name server, or by adding an authoritative DNS record (A record for IPv4 or AAAA record for IPv6) to the NetScaler configuration.

When you configure services with domain names instead of IPs, if you change the IP address of a server in your load balancing setup, the name server resolves the domain name to the new IP address. The monitor runs a health check on the new IP address, and updates the service IP address only when the IP address is found to be healthy.

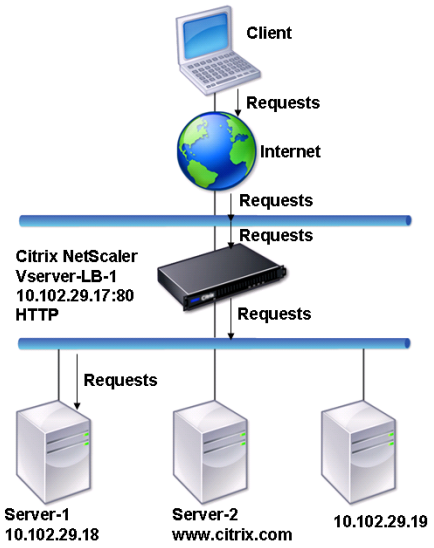
Note: When you change the IP address of a server, the corresponding service is marked down for the first client request. The name server resolves the service IP address to the changed IP address for the next request, and the service is marked UP.

Domain-name based services have the following restrictions:

- The maximum domain name length is 255 characters.
- The Maximum Client parameter is used to configure a service that represents the domain name-based server. For example, a maxClient of 1000 is set for the services bound to a virtual server. When the connection count at the virtual server reaches 2000, the DNS resolver changes the IP address of the services. However, because the connection counter on the service is not reset, the virtual server cannot take any new connections until all the old connections are closed.
- When the IP address of the service changes, persistence is difficult to maintain.
- If the domain name resolution fails due to a timeout, the appliance uses the old information (IP address).
- When monitoring detects that a service is down, the appliance performs a DNS resolution on the service (representing the domain name-based server) to obtain a new IP address.
- Statistics are collected on a service and are not reset when the IP address changes.
- If a DNS resolution returns a code of “name error” (3), the appliance marks the service down and changes the IP address to zero.

When the appliance receives a request for a service, it selects the target service. This way, the appliance balances load on your services. The following diagram describes the topology of a load balancing configuration that load balances a group of domain-name based servers (DBS).

Figure 1. Basic Load Balancing Topology for DBS Servers



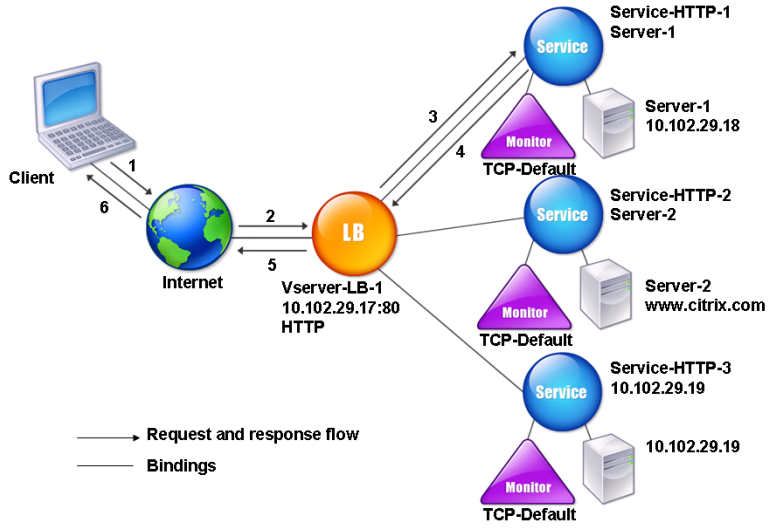
The services Service-HTTP-1, Service-HTTP-2, and Service-HTTP-3 are bound to the virtual server Vserver-LB-1. The vserver Vserver-LB-1 uses the least connection load balancing method to choose the service. The IP address of the service is resolved using the name server Vserver-LB-2.

The following table lists the names and values of the basic entities configured on the appliance.

Entity type	Name	IP address	Port	Protocol
Virtual Server	Vserver-LB-1	10.102.29.17	80	HTTP
	Vserver-LB-2	10.102.29.20	53	DNS
Servers	server-1	10.102.29.18	80	HTTP
	server-2	www.citrix.com	80	HTTP
Services	Service-HTTP-1	server-1	80	HTTP
	Service-HTTP-2	server-2	80	HTTP
	Service-HTTP-2	10.102.29.19	80	HTTP
Monitors	Default	None	None	None
Name Server	None	10.102.29.19	None	None

The following diagram shows the load balancing entities and the values of the parameters that need to be configured on the appliance.

Figure 2. Load Balancing DBS Servers Entity Model



To configure a basic load balancing setup, see Setting Up Basic Load Balancing. Create the services and virtual servers of type HTTP, and name the entities and set the parameters using the values described in the previous table.

You can add, remove, enable, and disable external name servers. You can create a name server by specifying its IP address, or you can configure an existing virtual server as the name server.

To add a name server by using the command line interface

At the command prompt, type:

add dns nameServer <dnsVserverName>

Example

add dns nameServer Vserver-LB-2

To add a name server by using the configuration utility

1. Navigate to Traffic Management > DNS > Name Servers.
2. In the details pane, click Add.
3. In the Create Name Server dialog box, select DNS Virtual Server.
4. In the DNS Virtual Server drop-down list, select the server name.
   Note: Click New if you want to create a new load balancing vserver. The Create Virtual Server (Load Balancing) dialog box appears.
5. Click Create, and then click Close.

You can also add an authoritative name server that resolves the domain name to an IP address.

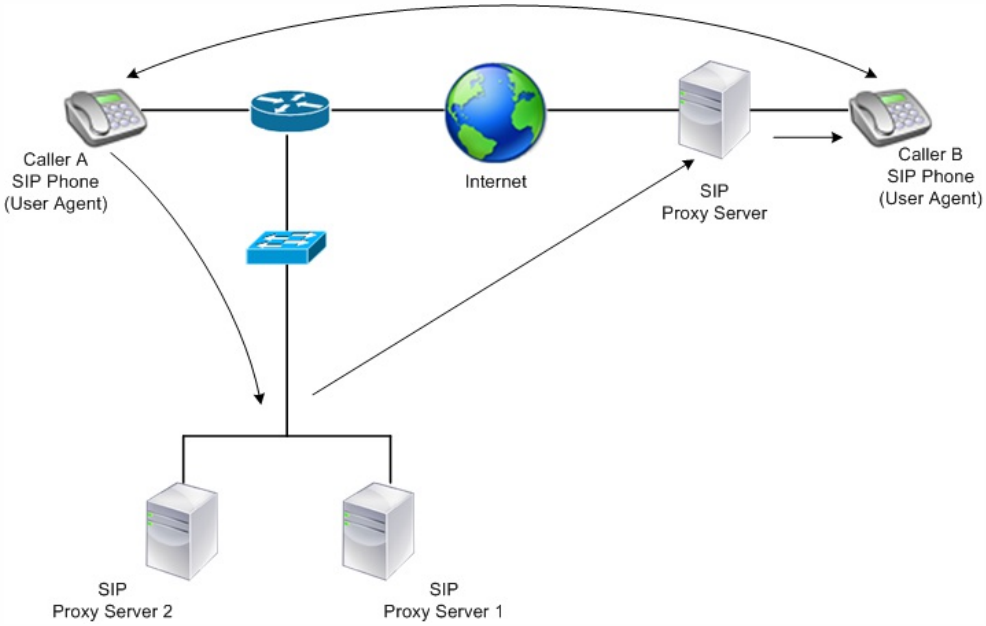
Load Balancing a Group of SIP Servers

Dec 09, 2014

The Session Initiation Protocol (SIP) is designed to initiate, manage, and terminate multimedia communications sessions. It has emerged as the standard for Internet telephony (VoIP). SIP messages can be transmitted over TCP or UDP. SIP messages are of two types: request messages and response messages.

The traffic in a SIP based communication system is routed through dedicated devices and applications (entities). In a multimedia communication session, these entities exchange messages. The following figure shows a basic SIP based communication system:

Figure 1. SIP Based Communication System



A NetScaler ADC enables you to load balance SIP messages over UDP or over TCP (including TLS). You can configure the NetScaler ADC to load balance SIP requests to a group of SIP proxy servers. To do so, you create a load balancing virtual server with the load balancing method and the type of persistence set to one of the following combinations:

- Call-ID hash load balancing method with no persistence setting
- Call-ID based persistence with least connection or round robin load balancing method
- Rule based persistence with least connection or round robin load balancing method

Also, by default, the NetScaler ADC appends RPORT to the via header of the SIP request, so that the server sends the response back to the source IP address and port from which the request originated.

Note: For load balancing to work, you must configure the SIP proxies so that they do not add private IP addresses or private domains to the SIP header/payload. SIP proxies must add to the SIP header a domain name that resolves to the IP address of the SIP virtual server. Also, the SIP proxies must communicate with a common database to share registration information.

Server Initiated Traffic

For SIP-server initiated outbound traffic, configure RNAT on the NetScaler ADC so that the private IP addresses used by the clients are translated into public IP addresses.

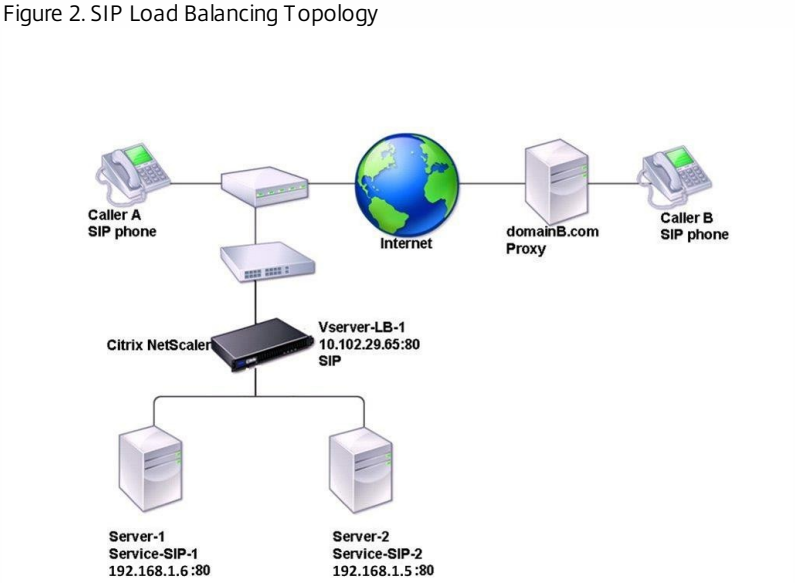
If you have configured SIP parameters that include the RNAT source or destination port, the appliance compares the values of the source and destination ports of the request packets with the RNAT source port and RNAT destination port. If one of the values matches, the appliance updates the VIA header with RPORT. The SIP response from the client then traverses the same path as the request.

For server-initiated SSL traffic, the NetScaler ADC uses a built-in certificate-key pair. If you want to use a custom certificate-key pair, bind the custom certificate-key pair to the NetScaler internal service named **nsrnatsip-127.0.0.1-5061**.

Support for Policies and Expressions

The NetScaler default expressions language contains a number of expressions that operate on Session Initiation Protocol (SIP) connections. These expressions can be bound only to SIP based (`sip_udp`, `sip_tcp` or `sip_ssl`) virtual servers, and to global bind points. You can use these expressions in content switching, rate limiting, responder, and rewrite policies.
For more information, see SIP Expressions.

Figure 2. SIP Load Balancing Topology

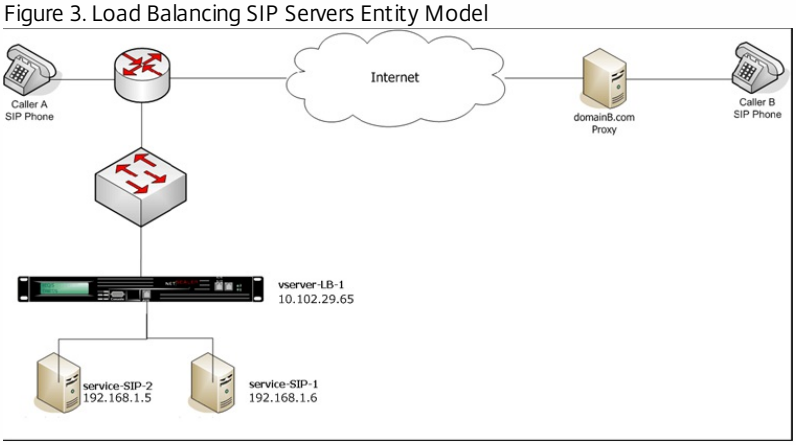


In the example, the services Service-SIP-1 and Service-SIP-2 are bound to the virtual server Vserver-LB-1. The following table lists the names and values of the entities that you need to configure on the appliance in inline mode (also called two-arm mode).

Entity type	Name	IP address	Port	Protocol
Virtual Server	Vserver-LB-1	10.102.29.65	80	SIP-UDP
Services	Service-SIP-1	192.168.1.6	80	SIP-UDP
	Service-SIP-2	192.168.1.5	80	SIP-UDP
Monitors	Default	None	80	SIP-UDP

The following diagram shows the load balancing entities and the values of the parameters to be configured on the appliance.

Figure 3. Load Balancing SIP Servers Entity Model



To configure a basic load balancing setup for SIP, see Setting Up Basic Load Balancing. You create services and virtual servers of type SIP-UDP, naming the entities and setting the parameters as described in the previous table. You must then configure RNAT.

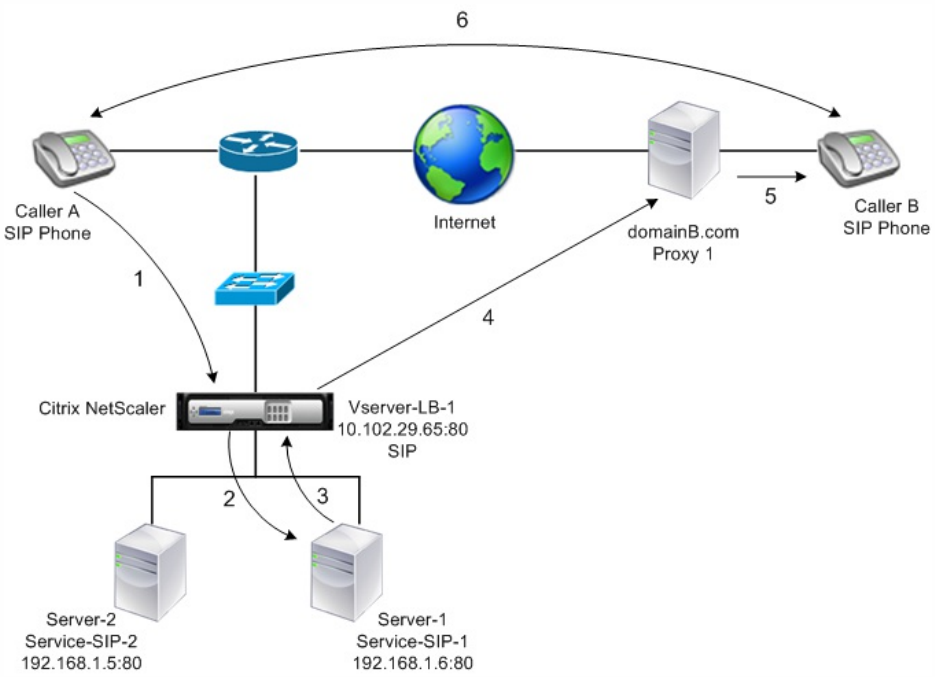
Configuring Load Balancing for SIP Signaling Traffic over TCP or UDP

The NetScaler ADC can load balance SIP servers that send requests over UDP or TCP, including TCP traffic secured by TLS. The ADC provides the following service types to load balance the SIP servers:

- SIP_UDP – Used when SIP servers send SIP messages over UDP.
- SIP_TCP – Used when SIP servers send SIP messages over TCP.
- SIP_SSL – Used to secure SIP signaling traffic over TCP by using SSL or TLS. The NetScaler ADC supports the following modes:
  - End-to-end TLS connection between the client, the ADC, and the SIP server.
  - TLS connection between the client and the ADC, and TCP connection between the ADC and the SIP server.
  - TCP connection between the client and the ADC, and TLS connection between the ADC and the SIP server.

The following figure shows the topology of a setup configured to load balance a group of SIP servers sending SIP messages over TCP or UDP.

Figure 4. SIP Load Balancing Topology



Entity type	Name	IP address	Port	Service type / Protocol
Virtual Server	Vserver-LB-1	10.102.29.65	80	SIP_UDP / SIP_TCP / SIP_SSL
Services	Service-SIP-1	192.168.1.6	80	SIP_UDP / SIP_TCP / SIP_SSL
	Service-SIP-2	192.168.1.5	80	SIP_UDP / SIP_TCP / SIP_SSL
Monitors	Default	None	80	SIP_UDP / SIP_TCP / SIP_SSL

Following is an overview of configuring basic load balancing for SIP traffic:

1. Configure services, and configure a virtual server for each type of SIP traffic that you want to load balance:
   - **SIP_UDP** – If you are load balancing the SIP traffic over UDP.
   - **SIP_TCP** – If you are load balancing the SIP traffic over TCP.
   - **SIP_SSL** – If you are load balancing and securing the SIP traffic over TCP.

   Note: If you use SIP_SSL, be sure to create an SSL certificate-key pair. For more information, see Adding a Certificate Key Pair.

2. Bind the services to the virtual servers.
3. If you want to monitor the states of the services with a monitor other than the default (**tcp-default**), create a custom monitor and bind it to the services. The NetScaler ADC provides two custom monitor types, **SIP-UDP** and **SIP-TCP**, for monitoring SIP services.
4. If using a SIP_SSL virtual server, bind an SSL certificate-key pair to the virtual server.
5. If you are using the NetSCaler ADC as the gateway for the SIP servers in your deployment, configure RNAT.
6. If you want to append RPORT to the SIP messages that are initiated from the SIP server, configure the SIP parameters.

To configure a basic load balancing setup for SIP traffic by using the command line interface

1. Create one or more services. At the command prompt, type:
   add service <name> <serverName> (SIP_UDP | SIP_TCP | SIP_SSL) <port>

   Example
   add service Service-SIP-UDP-1 192.0.2.5 SIP_UDP 80

2. Create as many virtual servers as necessary to handle the services that you created. The virtual server type must match the type of services that you will bind to it. At the command prompt, type:
   add lb vserver <name> <serverName> (SIP_UDP | SIP_TCP | SIP_SSL) <port>

   Example
   add lb vserver Vserver-LB-1 SIP_UDP 10.102.29.60 80

3. Bind each service to a virtual server. At the command prompt, type:
   bind lb vserver <name> <serverName>

   Example
   bind lb vserver Vserver-LB-1 Service-SIP-UDP-1

4. (Optional) Create a custom monitor of type SIP-UDP or SIP-TCP, and bind the monitor to the service. At the command prompt, type:
   add lb monitor <monitorName> <monitorType> [<interval>]

   bind lb monitor <monitorName> <ServiceName>

   Example
   add lb monitor mon1 sip-UDP -sipMethod REGISTER -sipuRI sip:mon@test.com -sipregURI sip:mon@test.com -respcode 200

   bind monitor mon1 Service-SIP-UDP-1

5. If you created a SIP_SSL virtual server, bind an SSL certificate key pair to the virtual server. At the command prompt, type: At the command prompt, type:
   bind ssl vserver <vServerName> -certkeyName <certificate-KeyPairName> -CA –skipCAName

   Example
   bind ssl vserver Vserver-LB-1 -certkeyName CertKey-SSL-1

6. Configure RNAT as required by your network topology. At the command prompt, type one of the following commands to create, respectively, an RNAT entry that uses a network address as the condition and a MIP or SNIP as the NAT IP address, an RNAT entry that uses a network address as the condition and a unique IP address as the NAT IP address, an RNAT entry that uses an ACL as the condition and a MIP or SNIP as the NAT IP address, or an RNAT entry that uses an ACL as a condition and a unique IP address as the NAT IP address:
   set rnat <IPAddress> <netmask>

   set rnat <IPAddress> <netmask> -natip <NATIPAddress>

   set rnat <aclname> [-redirectPort <port>]

   set rnat <aclname> [-redirectPort <port>] -natIP <NATIPAddress>

   Example
   set rnat 192.168.1.0 255.255.255.0 -natip 10.102.29.50

If you want to use a custom certificate-key pair, bind the custom certificate-key pair to the NetScaler internal service named **nsrnatsip-127.0.0.1-5061**.

add ssl certKey <certkeyName> -cert <string> [-key <string>]

bind ssl service <serviceName> -certkeyName <string>

Example

add ssl certKey c1 -cert cert.epm -key key.ky

bind ssl service nsrnatsip-127.0.0.1-5061 -certkeyName c1

7. If you want to append RPORT to the SIP messages that the SIP server initiates, type the following command at the command prompt: set lb sipParameters -rnatSrcPort <rnatSrcPort> -rnatDstPort<rnatDstPort> -retryDur <integer> -addRportVip <addRportVip> -sip503RateThreshold <sip503_rate_threshold_value>

Sample Configuration for load balancing the SIP traffic over UDP

> add service service-UDP-1 10.102.29.5 SIP_UDP 80

Done

> add lb vserver vserver-LB-1 SIP_UDP 10.102.29.60 80

Done

> bind lb vserver vserver-LB-1 service-UDP-1

Done

> add lb mon mon1 sip-udp -sipMethod REGISTER -sipuRI sip:mon@test.com -sipregURI sip:mon@test.com -respcode 200

Done

> bind mon mon1 service-UDP-1

Done

> set rnat 192.168.1.0 255.255.255.0

Done

> set lb sipParameters -rnatSrcPort 5060 -rnatDstPort 5060 -retryDur 1000 -addRportVip ENABLED -sip503RateThreshold 1000

Done

Sample Configuration for load balancing the SIP traffic over TCP

> add service service-TCP-1 10.102.29.5 SIP_TCP 80

Done

> add lb vserver vserver-LB-1 SIP_TCP 10.102.29.60 80

Done

> bind lb vserver vserver-LB-1 service-TCP-1

Done

> add lb mon mon1 sip-tcp -sipMethod REGISTER -sipuRI sip:mon@test.com -sipregURI sip:mon@test.com -respcode 200

Done

```
> bind mon mon1 service-TCP-1

Done

> set rnat 192.168.1.0 255.255.255.0

Done

> set lb sipParameters -rnatSrcPort 5060 -rnatDstPort 5060 -retryDur 1000 -addRportVip ENABLED -sip503RateThreshold
1000

Done
```

Sample Configuration for load balancing and securing SIP traffic over TCP

```
> add service service-SIP-SSL-1 10.102.29.5 SIP_SSL 80

Done

> add lb vserver vserver-LB-1 SIP_SSL 10.102.29.60 80

Done

> bind lb vserver vserver-LB-1 service-SIP-SSL

Done

> add lb mon mon1 sip-tCP -sipMethod REGISTER -sipuRI sip:mon@test.com -sipregURI sip:mon@test.com -respcode 200

Done

> bind mon mon1 service-SIP-SSL

Done

> bind ssl vserver Vserver-LB-1 -certkeyName CertKey-SSL-1

Done

> set rnat 192.168.1.0 255.255.255.0

Done

> set lb sipParameters -rnatSrcPort 5060 -rnatDstPort 5060 -retryDur 1000 -addRportVip ENABLED -sip503RateThreshold
1000

Done
```

To configure a basic load balancing setup for SIP traffic by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers, and add a virtual server of type SIP_UDP, SIP_TCP, or SIP_SSL.
2. Click the Service section, and add a service of type SIP_UDP, SIP_TCP, or SIP_SSL.
3. (Optional) Click the Monitor section, and add a monitor of type: SIP-UDP or SIP-TCP.
4. Bind the monitor to the service, and bind the service to the virtual server.
5. If you created a SIP_SSL virtual server, bind an SSL certificate key pair to the virtual server. Click the Certificates section, and bind a certificate key pair to the virtual server.
6. Configure RNAT as required by your network topology. To configure RNAT:
   1. Navigate to System > Network > Routes.
   2. On the Routes page, click the RNAT tab.
   3. In the details pane, click Configure RNAT.
   4. In the Configure RNAT dialog box, do one of the following:
      - If you want to use the network address as a condition for creating an RNAT entry, click Network and set the following parameters:
        - Network

- Netmask
- If you want to use an extended ACL as a condition for creating an RNAT entry, click ACL and set the following parameters:
  - ACL Name
  - Redirect Port

5. To set a MIP or SNIP address as a NAT IP address, skip to step 7.
6. To set a unique IP address as a NAT IP, in the Available NAT IP (s) list, select the IP address that you want to set as the NAT IP, and then click Add. The NAT IP you selected appears in the Configured NAT IP(s) list.
7. Click Create, and then click Close.

If you want to use a custom certificate-key pair, bind the custom certificate-key pair to the NetScaler internal service named **nsrnatsip-127.0.0.1-5061**. To bind the pair:

1. Navigate to Traffic Management > Load Balancing > Services and click the Internal Services tab.
2. Select nsrnatsip-127.0.0.1-5061 and click **Edit**.
3. Click the **Certificates** section and bind a certificate key pair to the internal service.

7. If you want to append RPORT to the SIP messages that the SIP server initiates, configure the SIP parameters. Navigate to Traffic Management > Load Balancing and click Change SIP settings, set the various SIP parameters.

SIP Expression and Policy Example: Compression Enabled in Client Requests

A NetScaler ADC cannot process compressed client SIP requests, so the client SIP request fails.

You can configure a responder policy that intercepts the SIP NEGOTIATE message from the client and looks for the compression header. If the message includes a compression header, the policy responds with "400 Bad Request," so that the client resends the request without compressing it.

At the command prompt, type the following commands to create the responder policy:

```
> add responder action sipaction1 respondwith q{"SIP/2.0 400 Bad Request\r\n\r\n"}

Done.

> add responder policy sippol1

> add responder policy sippol1 "SIP.REQ.METHOD.EQ(\"NEGOTIATE\")&&SIP.REQ.HEADER(\"Compression\").EXISTS" sipaction1
```

To configure RNAT by using the command line interface

At the command prompt, type:

set rnat<network> <netmask>
Example

```
set rnat 192.168.1.0 255.255.255.0
```

To configure RNAT by using the configuration utility

1. Navigate to System > Network > Routes.
2. On the Routes page, click the RNAT tab.
3. In the details pane, click Configure RNAT.
4. In the Configure RNAT dialog box, do one of the following:
   - If you want to use the network address as a condition for creating an RNAT entry, click Network and set the following parameters:
     - Network
     - Netmask
   - If you want to use an extended ACL as a condition for creating an RNAT entry, click ACL and set the following parameters:
     - ACL Name
     - Redirect Port
5. To set a MIP or SNIP address as a NAT IP address, skip to step 7.
6. To set a unique IP address as a NAT IP, in the Available NAT IP (s) list, select the IP address that you want to set as the NAT IP, and then click Add. The NAT IP you selected appears in the Configured NAT IP(s) list.

7. Click Create, and then click Close.

After you configure RNAT, the appliance sends SIP responses to the IP address and port that the client uses to send the request. The appliance also adds the RPORT tag in the VIA header of the message. The appliance compares the values of the source and destination ports of the request packets with the RNAT source port and RNAT destination port. If one of the values matches, the appliance updates the VIA header with the RPORT setting.

You must enable this setting when RPORT is not configured on either client.

To configure SIP parameters by using the command line interface

At the command prompt, type:

set lb sipParameters -rnatSrcPort <rnatSrcPort> -rnatDstPort<rnatDstPort> -retryDur <integer> -addRportVip <addRportVip> -sip503RateThreshold <sip503_rate_threshold_value>

Example

```
set lb sipParameters -rnatSrcPort 5060 -rnatDstPort 5060 -retryDur 1000 -addRportVip ENABLED -sip503RateThreshold 1000
```

To configure SIP parameters by using the configuration utility

1. Navigate to Traffic Management > DNS > Name Servers.
2. Create a DNS name server of type DNS Virtual Server, and select a server from the DNS Virtual Server list.

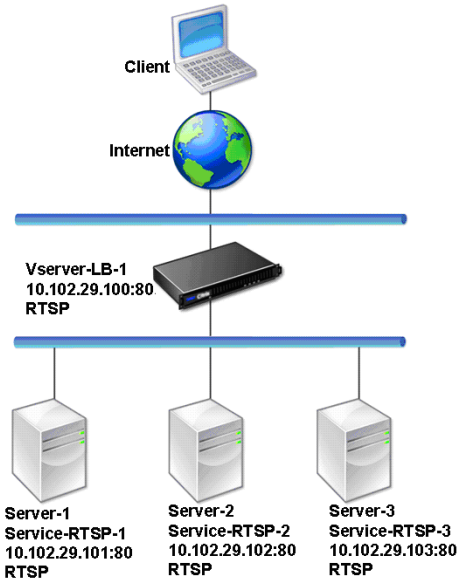
1. Navigate to Traffic Management > Load Balancing.
2. On the Load Balancing landing page, under Settings, click Change SIP settings.
3. In the Set SIP Parameters dialog box, set values for the following parameters:
   - RNAT Source Port
   - RNAT Destination Port
   - Retry Duration (secs)
   - SIP503 Rate Threshold
4. Select Enable Add Rport VIP.
5. Click OK.

Load Balancing RTSP Servers

Nov 12, 2013

The NetScaler appliance can balance load on RTSP servers to improve the performance of audio and video streams over networks. The following diagram describes the topology of an load balancing setup configured to load balance a group of RTSP servers.

Figure 1. Load Balancing Topology for RTSP

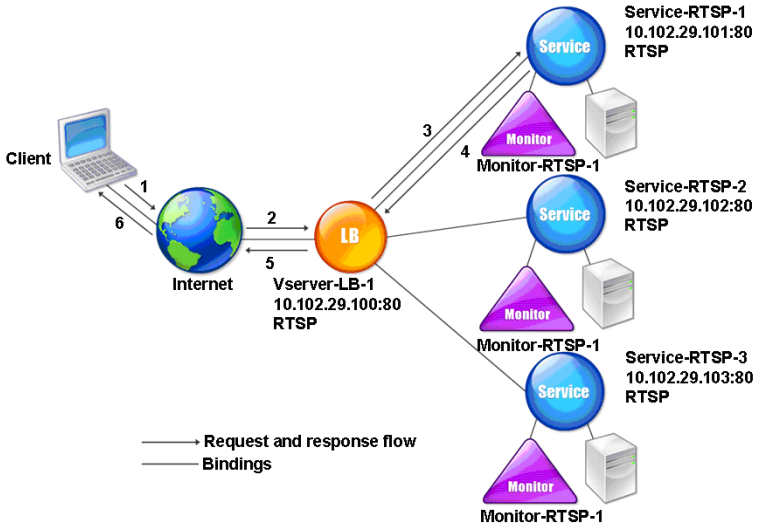


In the example, the services Service-RTSP-1, Service-RTSP-2, and Service-RTSP-3 are bound to the virtual server Vserver-LB-1. The following table lists the names and values of the example entities.

Entity type	Name	IP address	Port	Protocol
Virtual Server	Vserver-LB-1	10.102.29.100	554	RTSP
Services	Service-RTSP-1	10.102.29.101	554	RTSP
	Service-RTSP-2	10.102.29.102	554	RTSP
	Service-RTSP-3	10.102.29.103	554	RTSP
Monitors	Monitor-RTSP-1	None	554	RTSP

The following diagram shows the load balancing entities used in RTSP configuration.

Figure 2. Load Balancing RTSP Servers Entity Model



To configure a basic load balancing setup for RTSP servers, see Setting Up Basic Load Balancing. Create services and virtual servers of type RTSP. When you configure a basic load balancing setup, the default TCP-default monitor is bound to the services. To bind an RTSP monitor to these services, see Binding Monitors to Services. The following procedure describes how create a monitor that checks RTSP servers.

To configure RTSP monitors by using the command line interface

At the command prompt, type:

add lb monitor <monitorName> <type>

Example

add lb monitor Monitor-RTSP-1 RTSP

To configure RTSP monitors by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, in the Name and Interval text boxes, type the name and probing interval of a monitor.
4. In the Type list, select the type of the monitor.
5. Click Create, and then click Close.

Load Balancing of Remote Desktop Protocol (RDP) Servers

Nov 12, 2013

Remote Desktop Protocol (RDP) is a multichannel-capable protocol that allows for separate virtual channels for carrying presentation data, serial device communication, licensing information, highly encrypted data (keyboard and mouse activity), and so on.

RDP is used for providing a graphical user interface to another computer on the network. RDP is used with Windows terminal servers for providing fast access with almost real-time transmission of mouse movements and key presses even over low-bandwidth connections.

When multiple terminal servers are deployed to provide remote desktop services, the NetScaler appliance provides load balancing of the terminal servers (Windows 2003 and 2008 Server Enterprise Editions). In some cases, a user who is accessing an application remotely may want to leave the application running on the remote machine but shut down the local machine. The user therefore closes the local application without logging out of the remote application. After reconnecting to the remote machine, the user should be able to continue with the remote application. To provide this functionality, the NetScaler RDP implementation honors the routing token (cookie) set by the Terminal Services Session Directory or Broker so that the client can reconnect to the same terminal server to which it was connected previously. The Session Directory, implemented on Windows 2003 Terminal Server, is referred to as Broker on Windows 2008 Terminal Server.

When a TCP connection is established between the client and the load balancing virtual server, the NetScaler applies the specified load balancing method and forwards the request to one of the terminal servers. The terminal server checks the session directory to determine whether the client has a session running on any other terminal server in the domain.

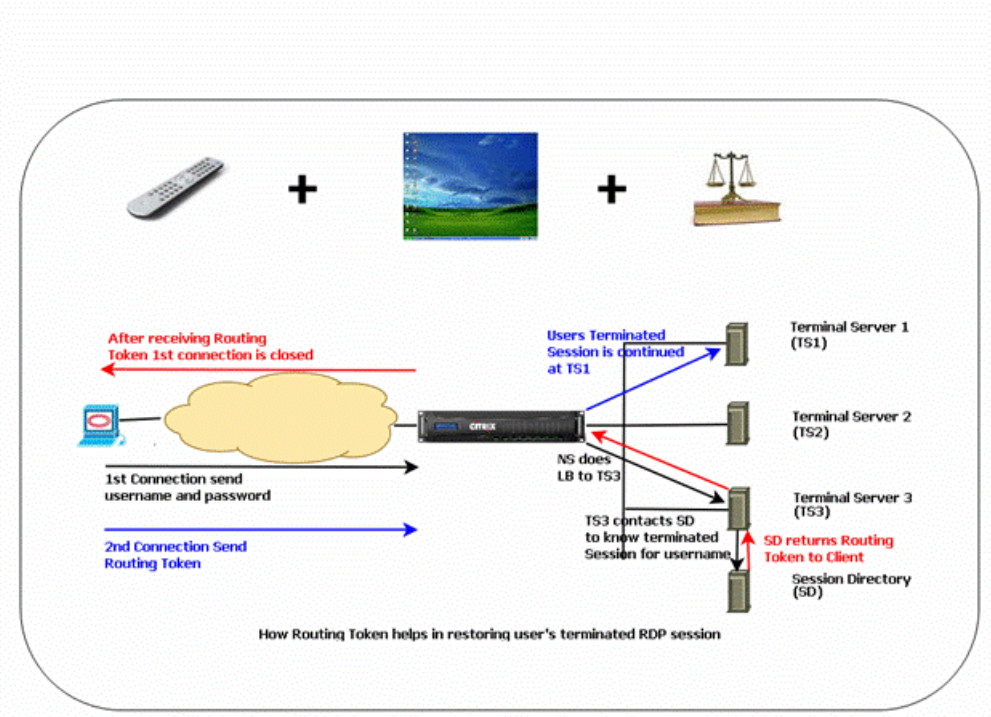
If there is no active session on any other terminal server, the terminal server responds by serving the client request, and the NetScaler forwards the response to the client.

If there is an active session on any other terminal server, the terminal server that receives the request inserts a cookie (referred to as routing token) with the details of the active session and returns the packets to the NetScaler, which returns the packet to the client. The server closes the connection with the client. When the client retries to connect, the NetScaler reads the cookie information and forwards the packet to the terminal server on which the client has an active session.

The user on the client machine experiences a continuation of the service and does not have to take any specific action.

Note: The Windows Session Directory feature requires the Remote Desktop client that was first released with Windows XP. If a session with a Windows 2000 or Windows NT 4.0 Terminal Server client is disconnected and the client reconnects, the server with which the connection is established is selected by the load balancing algorithm.
The following diagram describes RDP load balancing.

Figure 1. Load Balancing Topology for RDP



Note: When an RDP service is configured, persistence is automatically maintained by using a routing token. You need not enable persistence explicitly.
Ensure that the disconnected RDP sessions are cleared on the terminal servers at the backend to avoid flapping between two terminal servers when an RDP session is disconnected without logging out. For more information, see http://technet.microsoft.com/en-us/library/cc758177(WS.10).aspx#BKMK_2

When you add an RDP service, by default, NetScaler adds a monitor of the type TCP and binds it to the service. The default monitor is a simple TCP monitor that checks whether or not a listening process exists at the 3389 port on the server specified for the RDP service. If there is a listening process at 3389, NetScaler marks this service as UP and if there is no listening process, it marks the service as DOWN.

For more efficient monitoring of an RDP service, in addition to the default monitor, you can configure a script monitor that is meant for the RDP protocol. When you configure the scripting monitor, the NetScaler opens a TCP connection to the specified server and sends an RDP packet. The monitor marks the service as UP only if it receives a confirmation of the connection from the physical server. Therefore, from the scripting monitor, the NetScaler can know whether the RDP service is ready to service a request.

The monitor is a user-type monitor and the script is located on the NetScaler at /nsconfig/monitors/nsrdp.pl. When you configure the user monitor, the NetScaler runs the script automatically. To configure the scripting monitor, add the monitor and bind it to the RDP service.

To configure RDP load balancing, create services of type RDP and bind them to an RDP virtual server.

To configure RDP load balancing services by using the command line interface

At the command prompt, type the following commands to configure an RDP load balancing setup and verify the configuration:

add service <name>@ <serverName> <serviceType> <port>

Note: Repeat the above command to add more services.

Example

```
> add service ser1 10.102.27.182 RDP 3389
Done
> add service ser2 10.102.27.183 RDP 3389
Done
>show service ser1
ser1 (10.102. 27.182:3389) - RDP
      State: UP
…
  Server Name: 10.102.27.182
  Server ID : 0    Monitor Threshold : 0
      Down state flush: ENABLED
…
1)     Monitor Name: tcp-default
            State: UP      Weight: 1
…
            Response Time: 4.152 millisec
 Done
```

To configure RDP load balancing services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box:
   - Service Name*—serviceName
   - Protocol*—serviceType
   - Server*—serverName
   - Port*—port

   *A required parameter
4. Click Create.
5. Create all the RDP services to be load balanced.
6. From the services pane, open the service you added, and verify the addition.

To configure an RDP load balancing virtual server by using the command line interface

At the command prompt, type the following commands to configure an RDP load balancing virtual server and verify the configuration:

- add lb vserver <name>@ <serviceType> <ipAddress> <port>
- bind lb vserver <name>@ <serviceName>

  Bind all the RDP services to be load balanced to the virtual server.

Example

This example has two RDP services bound to the RDP virtual server.

```
> add lb vs v1 rdP 10.102.27.186 3389
Done

> bind lb vs v1 ser1
service "ser1" bound
> bind lb vs v1 ser2
service "ser2" bound
Done

>sh lb vs v1
v1 (10.102.27.186:3389) - RDP   Type: ADDRESS
State: UP
...
No. of Bound Services :  2 (Total)      2 (Active)
Configured Method: LEASTCONNECTION
  Current Method: Round Robin, Reason: A new service is bound
Mode: IP
Persistence: NONE
  L2Conn: OFF

1) ser1 (10.102.27.182: 3389) - RDPState: UP   Weight: 1
2) ser2 (10.102.27.183: 3389) - RDPState: UP   Weight: 1
Done
```

To configure an RDP load balancing virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters, which correspond to parameters described in "Parameters for configuring a virtual server" as shown:
   - Name*—vServerName
   - IP Address*—ipAddress
   - Protocol*—serviceType
   - Port*—port

   *A required parameter
4. In the Services tab, select the services to be bound to the virtual server by checking the service names.
5. Click Create.
6. In the Load Balancing Virtual Servers pane, select the RDP virtual server you configured, and then click Open to verify the configuration.

To configure a scripting monitor for RDP services by using the command line interface

At the command prompt, type the following commands:

- add lb monitor <monitorName> USER -scriptName nsrdp.pl
- bind lb monitor <monitorName> <rdpServiceName>

Example

```
add service ser1 10.102.27.182 RDP 3389
add lb monitor RDP_MON USER -scriptName nsrdp.pl
bind lb monitor RDP_MON ser1
```

To configure a scripting monitor for RDP services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify a name for the RDP monitor.
4. In the Type drop-down list, select USER.
5. On the Special Parameters tab, for the Script Name, click Browse and select nsrdp.pl from the default location.
6. Click Create.
7. From the Monitors pane, open the monitor you added, and verify the addition.
8. Navigate to Traffic Management > Load Balancing > Services.
9. In the details pane, select the RDP service, and then click Open.
10. In the Configure Service dialog box, select the RDP scripting monitor that you added and click Add.
11. Click OK.

Use Case 1: Configuring Rule Based Persistence Based on a Name-Value Pair in a TCP Byte Stream

Jun 08, 2015

Some protocols transmit name-value pairs in a TCP byte stream. The protocol in the TCP byte stream in this example is the Financial Information eXchange (FIX) protocol. In its traditional, non-XML implementation, the FIX protocol enables two hosts communicating over a network to exchange business or trade-related information as a list of name-value pairs (called "FIX fields"). The field format is <tag>=<value><delimiter>. This traditional tag-value format makes the FIX protocol ideal for the use case.

The tag in a FIX field is a numeric identifier that indicates the meaning of the field. For example, the tag 35 indicates the message type. The value after the equal sign holds a specific meaning for the given tag and is associated with a data type. For example, a value of A for the tag 35 indicates that the message is a logon message. The delimiter is the nonprinting "Start of Header" (SOH) ASCII character (0x01), which is the caret symbol (^). Each field is also assigned a name. For example, the field with tag 35 is the msgType field. Following is an example of a logon message.

```
8=FIX.4.1 9=61 35=A 49=INVMGR 56=BRKR 34=1 52=20000426-12:05:06 98=0 108=30 10=157
```

Your choice of persistence type for a tag-value list such as the one shown above is determined by the options that are available to you for extracting a particular string from the list. Token-based persistence methods require you to specify the offset and length of the token that you want to extract from the payload. The FIX protocol does not allow you to do that, because the offset of a given field and the length of its value can vary from one message to another (depending on the message type, the preceding fields, and the lengths of the preceding values) and from one implementation to another (depending on whether custom fields have been defined). Such variations make it impossible to predict the exact offset of a given field or to specify the length of the value that is to be extracted as the token. In this case, therefore, rule based persistence is the preferred persistence type.

Assume that a virtual server fixlb1 is load balancing TCP connections to a farm of servers hosting instances of a FIX-enabled application, and that you want to configure persistence for connections on the basis of the value of the SenderCompID field, which identifies the firm sending the message. The tag for this FIX field is 49 (shown in the earlier logon message example).

To configure rule based persistence for the load balancing virtual server, set the persistence type for the load balancing virtual server to RULE and configure the rule parameter with an expression. The expression must be one that extracts the portion of the TCP payload in which you expect to find the SenderCompID field, typecasts the resulting string to a name-value list based on the delimiters, and then extracts the value of the SenderCompID field (tag 49), as follows:

```
set lb vserver fixlb1 -persistenceType RULE -rule
"CLIENT.TCP.PAYLOAD(300).TYPECAST_NVLIST_T('=','^').VALUE(\"49\")"
```

Note: Backslash characters have been used in the expression because this is a CLI command. If you are using the configuration utility, do not enter the backslash characters.
If the client sends a FIX message that contains the name-value list in the earlier logon message example, the expression extracts the value INVMGR, and the NetScaler appliance creates a persistence session based on this value.

The argument to the PAYLOAD() function can be as large as you deem is necessary to include the SenderCompID field in the string extracted by the function. Optionally, you can use the SET_TEXT_MODE(IGNORECASE) function if you want

the appliance to ignore case when extracting the value of the field, and the HASH function to create a persistence session based on a hash of the extracted value. The following expression uses the SET_TEXT_MODE(IGNORECASE) and HASH functions:

```
CLIENT.TCP.PAYLOAD(500).TYPECAST_NVLIST_T('=','^').SET_TEXT_MODE(IGNORECASE).VALUE("49").HASH
```

Following are more examples of rules that you can use to configure persistence for FIX connections (replace <tag> with the tag of the field whose value you want to extract):

- To extract the value of any FIX field in the first 300 bytes of the TCP payload, you can use the expression CLIENT.TCP.PAYLOAD(300).BEFORE_STR("^").AFTER_STR("<tag>=").
- To extract a string that is 20 bytes long at offset 80, cast the string to a name-value list, and then extract the value of the field that you want, use the expression CLIENT.TCP.PAYLOAD(100).SUBSTR(80,20).TYPECAST_NVLIST_T('=','^').VALUE("<tag>").
- To extract the first 100 bytes of the TCP payload, cast the string to a name-value list, and extract the value of the third occurrence of the field that you want, use the expression CLIENT.TCP.PAYLOAD(100).TYPECAST_NVLIST_T('=','^').VALUE("<tag>",2).
  Note: If the second argument that is passed to the VALUE() function is n, the appliance extracts the value of the $(n+1)^{th}$ instance of the field because the count starts from zero (0).

Following are more examples of rules that you can use to configure persistence. Only the payload-based expressions can evaluate data being transmitted through the FIX protocol. The other expressions are more general expressions for configuring persistence based on lower networking protocols.

- CLIENT.TCP.PAYLOAD(100)
- CLIENT.TCP.PAYLOAD(100).HASH
- CLIENT.TCP.PAYLOAD(100).SUBSTR(5,10)
- CLIENT.TCP.SRCPORT
- CLIENT.TCP.DSTPORT
- CLIENT.IP.SRC
- CLIENT.IP.DST
- CLIENT.IP.SRC.GET4
- CLIENT.IP.DST.GET4
- CLIENT.ETHER.SRCMAC.GET6
- CLIENT.ETHER.DSTMAC.GET5
- CLIENT.VLAN.ID

Use Case 2: Configuring Load Balancing in Direct Server Return Mode

Jun 08, 2015

Load balancing in direct server return (DSR) mode allows the server to respond to clients directly by using a return path that does not flow through the NetScaler appliance. In DSR mode, however, the appliance can continue to perform health checks on services. In a high-data volume environment, sending server traffic directly to the client in DSR mode increases the overall packet handling capacity of the appliance because the packets do not flow through the appliance.

DSR mode has the following features and limitations:

- It supports one-arm mode and inline mode.
- The appliance ages out sessions based on idle timeout.
- Because the appliance does not proxy TCP connections (that is it does not send SYN-ACK to the client), it does not completely shut out SYN attacks. By using the SYN packet rate filter, you can control the rate of SYNs to the server. To control the rate of SYNs, set a threshold for the rate of SYNs. To get protection from SYN attacks, you must configure the appliance to proxy TCP connections. However, that requires the reverse traffic to flow through the appliance.
- In a DSR configuration, the NetScaler appliance does not replace the load balancing virtual server's IP address with the destination server's IP address. Instead, it forwards packets to a service by using the server's MAC address, which it obtains from the monitor bound to the service. However, custom user monitors (monitors of type USER), which use scripts stored on the NetScaler appliance, do not learn a server's MAC address. If you use only custom monitors in a DSR configuration, for each request the virtual server receives, the appliance attempts to resolve the destination IP address to a MAC address (by sending ARP requests). Because the destination IP address is a virtual IP address owned by the NetScaler appliance, the ARP requests always resolve to the MAC address of the NetScaler interface. Consequently, all traffic received by the virtual server is looped back to the appliance. If you use user monitors in a DSR configuration, you must also configure another monitor of a different type (for example, a PING monitor) for the services, ideally with a longer interval between probes, so that the MAC address of the servers can be learned.

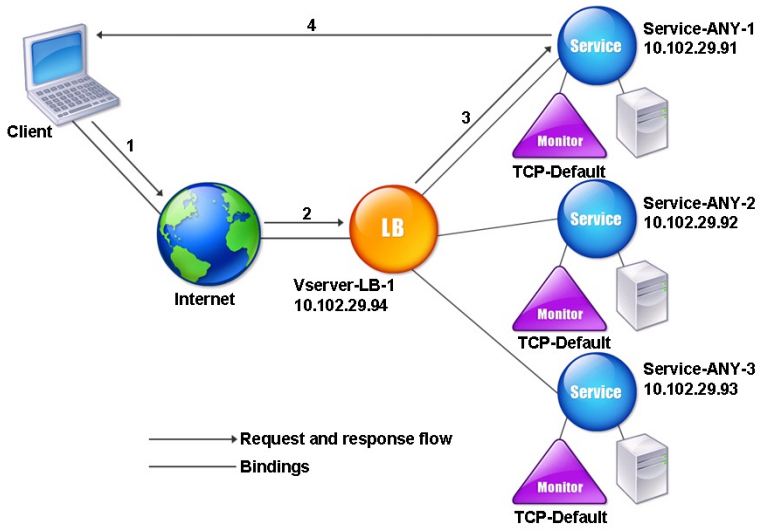
In the example scenario, the services Service-ANY-1, Service-ANY-2, and Service-ANY-3 are created and bound to the virtual server Vserver-LB-1. The virtual server load balances the client request to a service, and the service responds to clients directly, bypassing the NetScaler. The following table lists the names and values of the entities configured on the NetScaler in DSR mode.

Entity type	Name	IP address	Protocol
Virtual server	Vserver-LB-1	10.102.29.94	ANY
Services	Service-ANY-1	10.102.29.91	ANY
	Service-ANY-2	10.102.29.92	ANY
	Service-ANY-3	10.102.29.93	ANY

Entity type	Name	IP address	Protocol
Monitors	TCP	None	None

The following diagram shows the load balancing entities and values of the parameters to be configured on the appliance.

Figure 1. Entity Model for Load Balancing in DSR Model



For the appliance to function correctly in DSR mode, the destination IP in the client request must be unchanged. Instead, the appliance changes the destination MAC to that of the selected server. This setting enables the server to determine the client MAC address for forwarding requests to the client while bypassing the server. To enable the appliance to do this, you must enable MAC-based forwarding.

To enable MAC-based forwarding by using the command line interface

At the command prompt, type:

enable ns mode MACbasedforwarding

To enable MAC-based forwarding by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. On the Settings pane, under Modes and Features, click Configure modes.
3. In the Configure Modes dialog box, select the MAC Based Forwarding check box, and then click OK.
4. In the Enable/Disable Mode(s)? dialog box, click Yes.

Next, you configure a basic load balancing setup as described in Setting Up Basic Load Balancing, naming the entities and setting the parameters using the values described in the previous table.

After you configure the basic load balancing setup, you must customize it for DSR mode. To do this, you configure a supported load balancing method, such as the Source IP Hash method with a sessionless virtual server. You also need to set the redirection mode to allow the server to determine the client MAC address for forwarding responses and bypass the appliance.

After you configure the load balancing method and redirection mode, you need to enable the USIP mode on each service.

The service then uses the source IP address when forwarding responses.

To configure the load balancing method and redirection mode for a sessionless virtual server by using the command line interface

At the command prompt, type:

set lb vserver <vServerName> -lbMethod <LBMethodOption> -m <RedirectionMode> -sessionless <Value>

Example

```
set lb vserver Vserver-LB-1 -lbMethod SourceIPHash -m MAC -sessionless enabled
```

To configure the load balancing method and redirection mode for a sessionless virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server (for example, Vserver-LB-1), and then click Open.
3. On the Method and Persistence tab, under LB Method, select SOURCE IP Hash.
4. On the Advanced tab, under Redirection Mode, select MAC Based.
5. Select the Sessionless check box and click OK.

To configure a service to use source IP address by using the command line interface

At the command prompt, type:

set service <ServiceName> -usip <Value>

Example

```
set service Service-ANY-1 -usip yes
```

To configure a service to use source IP address by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. On the Services pane, click Service-ANY-1, and then click Open.
3. On the Advanced tab, under Settings, select the Use Source IP check box, and then click OK.
4. Repeat steps 1-5 for the services Service-ANY-2 and Service-ANY-3.

Some additional steps are required in certain situations, which are described in the succeeding sections.

Use Case 3: Configuring LINUX Servers in DSR Mode

Jun 08, 2015

The LINUX operating system requires that you set up a loopback interface with the NetScaler appliance virtual IP address (VIP) on each load balanced server in the DSR cluster.

To configure LINUX server in DSR mode

To create a loop back interface with the NetScaler appliance's VIP on each load balanced server, at the Linux OS prompt type the following commands:

ifconfig dummy0 up

ifconfig dummy0:0 inet <netscaler vip> netmask 255.255.255.255 up

echo 1 > /proc/sys/net/ipv4/conf/dummy0/arp_ignore

echo 2 > /proc/sys/net/ipv4/conf/dummy0/arp_announce

Then, run the software that re-maps the TOS id to VIP.

Note: Add the correct mappings to the software before running it. In the preceding commands, the LINUX server uses dummy0 to connect to the network. When you use this command, type the name of the interface that your LINUX server uses to connect to the network.

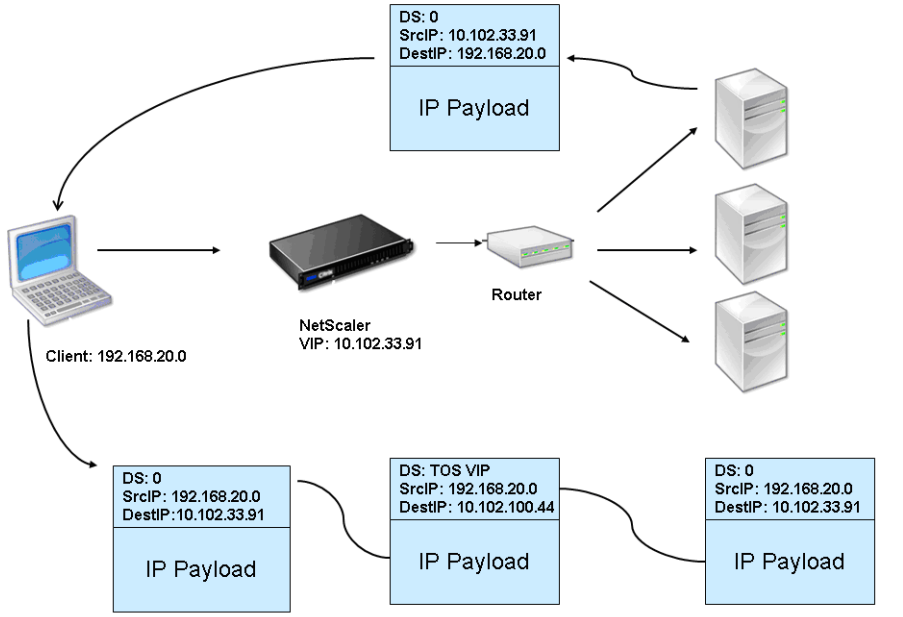
Use Case 4: Configuring DSR Mode When Using TOS

Jun 08, 2015

Differentiated services (DS), also known as TOS (Type of Service), is a field that is part of the TCP packet header. TOS is used by upper layer protocols for optimizing the path for a packet. The TOS information encodes the NetScaler appliance virtual IP address (VIP), and the load balanced servers extract the VIP from it.

In the following scenario, the appliance adds the VIP to the TOS field in the packet and then forwards the packet to the load balanced server. The load balanced server then responds directly to the client, bypassing the appliance, as illustrated in the following diagram.

Figure 1. The NetScaler Appliance in DSR mode with TOS



The TOS feature is specifically customized for a controlled environment, as described below:

- The environment must not have any stateful devices, such as stateful firewall and TCP gateways, in the path between the appliance and the load balanced servers.
- Routers at all the entry points to the network must remove the TOS field from all incoming packets to make sure that the load balanced server does not confuse another TOS field with that added by the appliance.
- Each server can have only 63 VIPs.
- The intermediate router must not send out ICMP error messages regarding fragmentation. The client will not understand the message, as the source IP address will be the IP address of the load balanced server and not the NetScaler VIP.
- TOS is valid only for IP-based services. You cannot use domain name based services with TOS.

In the example, Service-ANY-1 is created and bound to the virtual server Vserver-LB-1. The virtual server load balances the client request to the service, and the service responds to clients directly, bypassing the appliance. The following table lists the names and values of the entities configured on the appliance in DSR mode.

Entity Type	Name	IP Address	Protocol

Entity Type	Name	IP Address	Protocol
Virtual server	Vserver-LB-1	10.102.33.91	ANY
Services	Service-ANY-1	10.102.100.44	ANY
Monitors	PING	None	None

DSR with TOS requires that load balancing be set up on layer 3. To configure a basic load balancing setup for Layer 3, see Setting Up Basic Load Balancing. Name the entities and set the parameters using the values described in the previous table.

After you configure the load balancing setup, you must customize the load balancing setup for DSR mode by configuring the redirection mode to allow the server to decapsulate the data packet and then respond directly to the client and bypass the appliance.

After specifying the redirection mode, you can optionally enable the appliance to transparently monitor the server. This enables the appliance to transparently monitor the load balanced servers.

To configure the redirection mode for the virtual server by using the command line interface

At the command prompt, type:

set lb vserver <vServerName> -m <Value> -tosId <Value>
Example

set lb vserver Vserver-LB-1 -m TOS -tosId 3

To configure the redirection mode for the virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers pane, select the virtual server and click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, in Redirection Mode, click TOS Based.
4. In the TOS Id box, enter a value for the TOS ID.
5. Click OK.

To configure the transparent monitor for TOS by using the command line interface

At the command prompt, type:

add monitor <MonitorName> <Type> -destip <DestinationIP> -tos <Value> -tosId <Value>
Example

add monitor mon1 PING -destip 10.102.33.91 -tos Yes -tosId 3

To create the transparent monitor for TOS by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. On the Monitors pane, select the monitor (for example, tcp), and click Add.
3. In the Create Monitor dialog box, in the Name and Destination IP boxes, enter the monitor name and the destination IP address (for example, PING and 10.102.33.91).
4. In the Type list, select the type of monitor (for example, PING).
5. To configure the monitor for TOS, select the TOS check box.
6. In the TOS Id box, enter the same TOS ID that you had entered for the virtual server (for example, 3.)

7. Click OK.

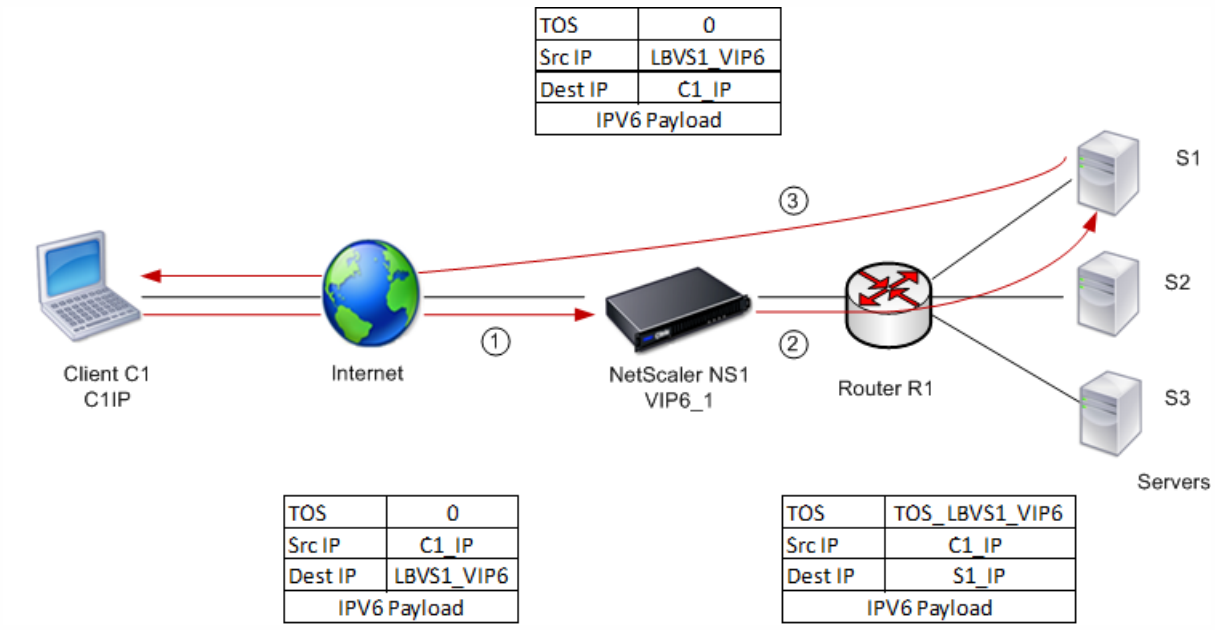
Use Case 5: Configuring Load Balancing in DSR Mode for IPv6 Networks by Using the TOS Field

Jun 08, 2015

You can configure load balancing in Direct Server Return (DSR) mode for IPv6 networks by using the Type of Service (TOS) field when the NetScaler appliance and the servers are in different networks.

Note: The TOS field is also called the Traffic Class field.
In DSR mode, when a client sends a request to a VIP6 address on a NetScaler appliance, the appliance forwards this request to the server by changing the destination IPv6 address of the packet to the IPv6 address of the server and sets an encoded value of the VIP6 address in the TOS (also called traffic class) field of the IPv6 header. You can configure the server to use the information in the TOS field to derive the VIP6 address from the encoded value, which is then used as source IP address in response packets. Response traffic directly goes to the client, bypassing the NetScaler.

Consider an example where a load balancing virtual server LBVS1, configured on a NetScaler appliance NS1, is used to load balance traffic across servers S1, S2, and S3. The NetScaler appliance NS1 and the servers S1, S2, and S3 are in different networks so router R1 is deployed between NS1 and the servers.



The following table lists the settings used in this example.

Entities	Name
IPv6 address of client C1	C1_IP (for reference purposes only)
Load balancing virtual server on NS1	LBVS1
IPv6 address of LBVS1	LBVS1_VIP6 (for references purpose only)

TOS value **Entities**	TOS_ LBVS1_VIP6 (for references purpose only) **Name**
Service for server S1 on NS1	SVC_S1
IPv6 address for server S1	S1_IP (for references purpose only)
Service for server S2 on NS1	SVC_S2
IPv6 address for server S1	S2_IP (for references purpose only)
Service for server S3 on NS1	SVC_S3
IPv6 address for server S1	S3_IP (for references purpose only)

Following is the traffic flow in the example scenario:

1. Client C1 sends a request to virtual server LBVS1.
2. LBVS1's load balancing algorithm selects server S1 and the appliance opens a connection to S1. NS1 sends the request to S1 with:
   - TOS field set to TOS_ LBVS1_VIP6.
   - Source IP address as C1_IP.
3. The server S1, on receiving the request, uses the information in the TOS field to derive the LBVS1_VIP6 address, which is the IP address of the virtual server LBVS1 on NS1. The server directly sends the response to C1, bypassing the NetScaler, with:
   - Source IP address set to the derivedLBVS1_VIP6 address so that the client communicates to the virtual server LBVS1 on NS1 and not to server S1.

To configure load balancing in DSR Mode using TOS, perform the following steps on the appliance:

1. Enable USIP mode globally.
2. Add the servers as services.
3. Configure a load balancing virtual server with a TOS value.
4. Bind the services to the virtual server.

To configure load balancing in DSR Mode using TOS by using the command line interface

At the command prompt, type:

- enable ns mode USIP
- add service <serviceName> <IP> <serviceType> <port>
  Repeat the above command as many times as necessary to add each server as a service on the NetScaler appliance.
- add lb vserver <name> <serviceType> <ip> <port> -m <redirectionMode> -tosId <positive_integer>
- bind lb vserver <vserverName> <serviceName>

To enable USIP mode by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.

2. In the Modes and Features group, click Configure Modes.
3. In the Configure Modes dialog box, select the Use Source IP check box.

To create services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, set the following parameters.
   - Service Name*
   - Server*
   - Protocol* (Select ANY from the drop-down list.)
   - Port*
4. Click **Create**.
5. Repeat steps 3-4 to create another service.
6. Click Close.
7. In the Services pane, select the services that you just configured and verify that the settings displayed at the bottom of the screen are correct.

To create a load balancing virtual server and bind services by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers pane, click Add.
3. In the Create Virtual Servers (Load Balancing) dialog box, select IP Address Based and then select IPv6.
4. Set the following parameters.
   - Name*
   - IP Address*
   - Protocol* (Select ANY from the drop-down list.)
   - Port*

   * A required parameter
5. On the Advanced tab, under Redirection Mode, click TOS Based. In the TOS Id box, enter a value.
6. Under the Services tab, in the Active column, select the check box for the service that you want to bind to the virtual server.
7. Click Create, and then click Close. In the Load Balancing Virtual Servers tab, select the virtual server that you just created, and verify that the settings displayed in the Details pane are correct.

Use Case 6: Configuring Load Balancing in DSR Mode by Using IP Over IP

Jun 08, 2015

You can configure your NetScaler appliance to use direct server return (DSR) mode across Layer 3 networks by using IP tunneling, also called *IP over IP* configuration. As with standard load balancing configurations for DSR mode, this allows servers to respond to clients directly instead of using a return path through the NetScaler appliance, improving response times and throughput. As with standard DSR mode, the NetScaler appliance monitors the servers and performs health checks on the application ports.

With IP over IP configuration, the NetScaler appliance and the servers do not need to be on the same Layer 2 subnet. Instead, the NetScaler appliance encapsulates the packets before sending them to the destination server. After the destination server receives the packets, it decapsulates the packets, and then sends its responses directly to the client.

To configure IP over IP DSR mode on your NetScaler appliance, you must do the following:

- **Create a load balancing virtual server.** Set the protocol to ANY and set the mode to IPTUNNEL.
- **Create services.** Create a service for each of your back-end applications. Bind the services that you created to the virtual server.

Configuring a Load Balancing Virtual Server

Updated: 2013-11-29

Configure a virtual server to handle requests to your applications. Assign a service type of ANY and set the forwarding method to IPTUNNEL. Optionally, configure the virtual server to operate in sessionless mode. You can configure any load balancing method that you want to use.

To create and configure a load balancing virtual server for IP over IP DSR by using the command line interface

At the command prompt type the following command to configure a load balancing virtual server for IP over IP DSR and verify the configuration:

- add lb vserver <name> serviceType <serviceType> IPAddress <ip> Port <port> -lbMethod <method> -m <ipTunnelTag> -sessionless <sessionless>
- show lb vserver <name>

Example

In the following example, we have selected the load balancing method as sourceIPhash and configured sessionless load balancing.

```
add lb vserver Vserver-LB-1 ANY 10.102.29.60 * -lbMethod SourceIPHash -m IPTUNNEL -sessionless enabled
```

To create and configure a load balancing virtual server for IP over IP DSR by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.

2. In the details pane, click Add.
3. In the Create Virtual Server dialog box, specify values for the following parameters:
   - Name*—name
   - Protocol*—protocol
   - IP address*—IPAddress
   - Port*—port

   * A required parameter
4. On the Advanced tab, under Redirection Mode, select IP Tunnel Based.
5. Click Create, and then click Close. The virtual server that you created now appears in the Virtual Servers pane.

Configuring Services for IP over IP DSR

Updated: 2013-11-29

After creating your load-balanced server, You must configure one service for each of your applications. The service handles traffic from the NetScaler appliance to those applications, and allows the NetScaler appliance to monitor the health of each application.

You assign a service type of ANY and configure it for USIP mode. Optionally, you can also bind a monitor of type IPTUNNEL to the service for tunnel-based monitoring.

To create and configure a service for IP over IP DSR by using the command line interface

At the command prompt, type the following commands to create a service and optionally, create a monitor and bind it to the service:

- add service <serviceName> <serverName> <serviceType> <port> -usip <usip>
- add monitor <monitorName> <monitorType> -destip <ip> -iptunnel <iptunnel>
- bind service <serviceName> -monitorName <monitorName>

Example

In the following example, we are creating a monitor of type IPTUNNEL:

```
add monitor mon-1 PING -destip 10.102.29.60 -iptunnel yes
add service Service-DSR-1 10.102.30.5 ANY * -usip yes
bind service Service-DSR-1 -monitorName mon-1
```

To configure a monitor by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Monitors.
2. In the details pane, click Add.
3. In the Create Monitor dialog box, specify values for the following parameters:
   - Monitor Name*—name
   - Type*—type
   - Destination IP—destip. Specify the IP address of the virtual server that you created earlier.

   * A required parameter
4. Select IP Tunnel.
5. Click Create, and then click Close.

To create and configure a service for IP over IP DSR by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, specify values for the following parameters:
   - Service Name*—name
   - Protocol*—type
   - Server*—IP
   - Port*—port

   * A required parameter
4. On the Monitors tab, from the Available list, select the monitor that you created earlier and add it to the Configured list.
5. On the Advanced tab, select Use Source IP.
6. Click Create, and then click Close.

To bind a service to a load balancing virtual server by using the command line interface

At the command prompt type the following command:

bind lb vserver <name> <serviceName>

Example

bind lb vserver Vserver-LB-1 Service-DSR-1

To bind a service to a load balancing virtual server by using the configuration utility

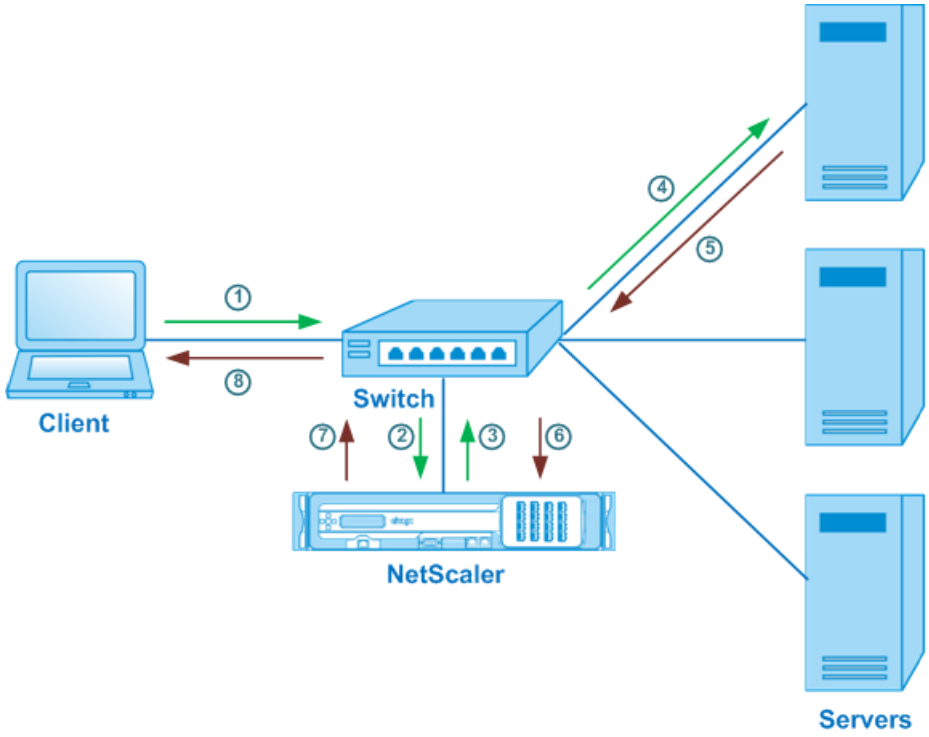
1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the load balancing virtual server that you created earlier, and then select Open.
3. In the Services tab, select the check box beside the name of the service(s) that you created earlier.
4. Click OK.

Use Case 7: Configuring Load Balancing in One-arm Mode

Feb 19, 2016

In a one-arm setup, you connect the NetScaler appliance to the network through a single VLAN. The appliance receives the request from the client on a single VLAN and it sends the request to the server on the same VLAN. This is one of the simplest deployment scenarios, where the router, the servers and the appliance are all connected to the same switch. Client requests at the switch are forwarded to the appliance, and the appliance uses the configured load balancing method to select the service, as is shown in the following diagram.

Figure: Load Balancing in One-Arm Mode



In the example scenario, the services Service-ANY-1, Service-ANY-2, and Service-ANY-3 are created and bound to the virtual server Vserver-LB-1. The virtual server load balances the client request to a service. The following table lists the names and values of the entities configured on the appliance in one-arm mode.

Entity type	Name	IP address	Protocol
Virtual server	Vserver-LB-1	10.102.29.94	ANY
Services	Service-ANY-1	10.102.29.91	ANY
	Service-ANY-2	10.102.29.92	ANY

Entity type	Name	IP address	Protocol
	Service-ANY-3	10.102.29.93	ANY
Monitors	TCP	None	None

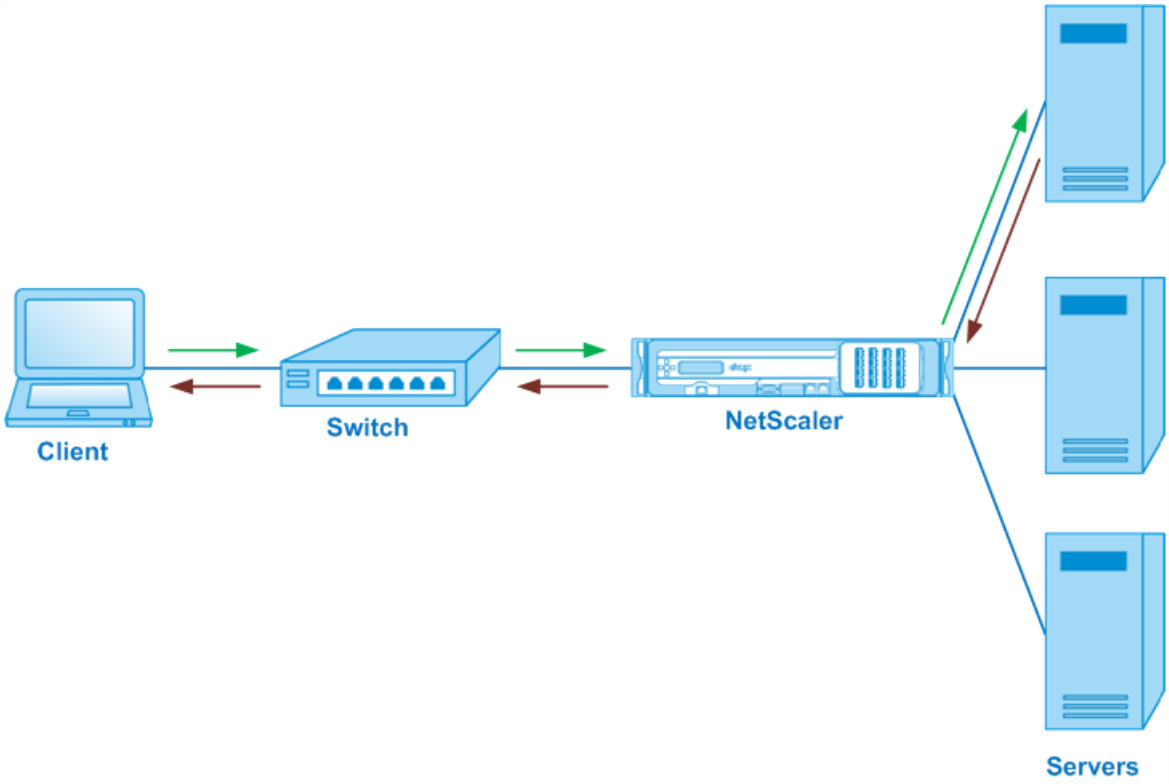
To configure a load balancing setup in one-arm mode, see "Setting Up Basic Load Balancing."

Use Case 8: Configuring Load Balancing in the Inline Mode

Feb 19, 2016

In an inline mode (also called two-arm mode) setup, you connect the NetScaler appliance to the network through multiple VLANs. The appliance receives the request from the client on one VLAN and it sends the request to the server on another VLAN. In the two-arm setup, the appliance is connected between the servers and the client. Client requests at the switch are forwarded to the appliance, and the appliance uses the configured load balancing method to select the service. This is shown in the following diagram.

Figure: Load Balancing in Inline Mode



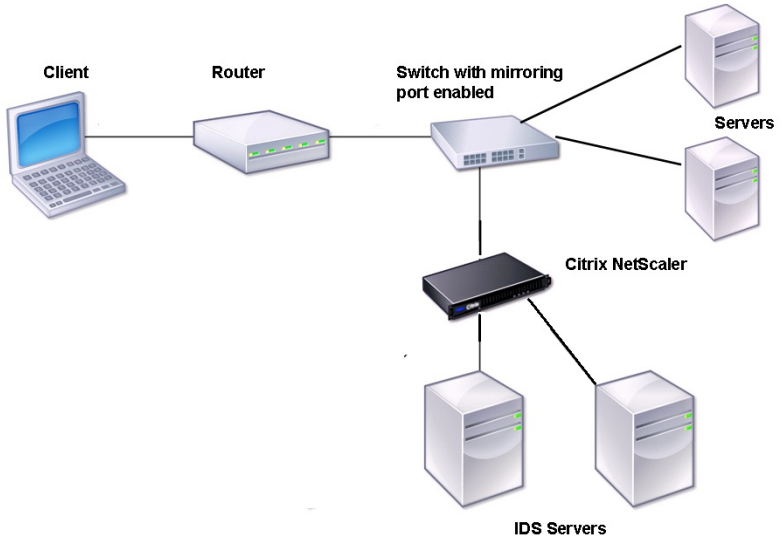
The configuration and the entity diagram for inline mode are the same as described in "Configuring Load Balancing in One-arm Mode."

Use Case 9: Load Balancing of Intrusion Detection System Servers

Jun 08, 2015

To enable the NetScaler appliance to support load balancing of intrusion detection system (IDS) servers, the IDS servers and clients must be connected through a switch that has port mirroring enabled. The client sends a request to the server. Because port mirroring is enabled on the switch, the request packets are copied or sent to the NetScaler appliance virtual server port. The appliance then uses the configured load balancing method to select an IDS server, as shown in the following diagram.

Figure 1. Topology of Load Balanced IDS Servers



Note: Currently, the appliance supports load balancing of passive IDS devices only.
As illustrated in the preceding diagram, the IDS load balancing setup functions as follows:

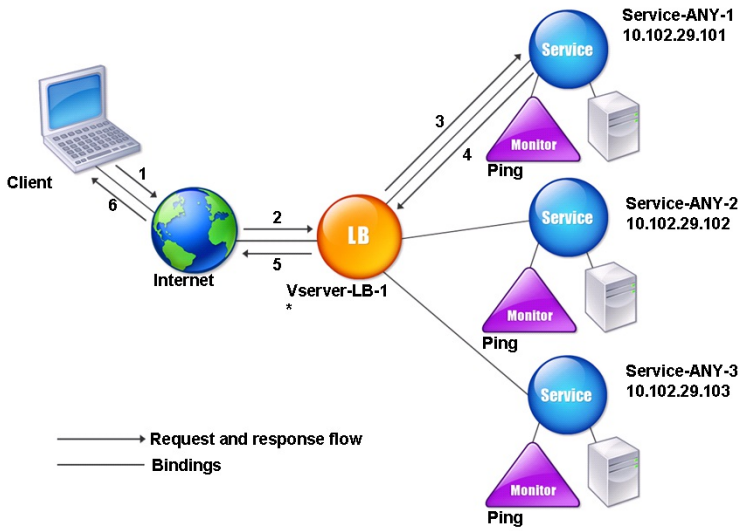
1. The client request is sent to the IDS server, and a switch with a mirroring port enabled forwards these packets to the IDS server. The source IP address is the IP address of the client, and the destination IP address is the IP address of the server. The source MAC address is the MAC address of the router, and the destination MAC address is the MAC address of the server.
2. The traffic that flows through the switch is mirrored to the appliance. The appliance uses the layer 3 information (source IP address and destination IP address) to forward the packet to the selected IDS server without changing the source IP address or destination IP address. It modifies the source MAC address and the destination MAC address to the MAC address of the selected IDS server.

Note: When load balancing IDS servers, you can configure the SRCIPHASH, DESTIPHASH, or SRCIPDESTIPHASH load balancing methods. The SRCIPDESTIPHASH method is recommended because packets flowing from the client to a service on the appliance must be sent to a single IDS server.
Suppose Service-ANY-1, Service-ANY-2, and Service-ANY-3 are created and bound to Vserver-LB-1. The virtual server balances the load on the services. The following table lists the names and values of the entities configured on the appliance.

Entity type	Name	IP address	Port	Protocol
Virtual server	Vserver-LB-1	*	*	ANY
Services	Service-ANY-1	10.102.29.101	*	ANY
	Service-ANY-2	10.102.29.102	*	ANY
	Service-ANY-3	10.102.29.103	*	ANY
Monitors	Ping	None	None	None

Note: You can use inline mode or one-arm mode for an IDS load balancing setup.
The following diagram shows the load balancing entities and values of the parameters to be configured on the appliance.

Figure 2. Entity Model for Load Balancing IDS Servers



To configure an IDS load balancing setup, you must first enable MAC-based forwarding. You must also disable layer 2 and layer 3 modes on the appliance.

To enable MAC-based forwarding by using the command line interface

At the command prompt, type:

enable ns mode <ConfigureMode>

Example

enable ns mode MAC

To enable MAC-based forwarding by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.
2. On the Settings landing page, under Modes and Features, click modes.
3. In the Configure Modes dialog box, select the MAC Based Forwarding check box, and then click OK.
4. In the Enable/Disable Feature(s)? dialog box, and then click Yes.

Next, see "Setting Up Basic Load Balancing", to configure a basic load balancing setup.

After you configure the basic load balancing setup, you must customize it for IDS by configuring a supported load balancing method (such as the SRCIPDESTIP Hash method on a sessionless virtual server) and enabling MAC mode. The appliance does not maintain the state of the connection and only forwards the packets to the IDS servers without processing them. The destination IP address and port remains unchanged because the virtual server is in the MAC mode.

To configure LB method and redirection mode for a sessionless virtual server by using the command line interface

At the command prompt, type:

set lb vserver <vServerName> -lbMethod <LBMethodOption> -m <RedirectionMode> -sessionless <Value>

Example

```
set lb vserver Vserver-LB-1 -lbMethod SourceIPDestIPHash -m MAC -sessionless enabled
```

To configure LB method and redirection mode for a sessionless virtual server by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the Load Balancing Virtual Servers pane, click the virtual server Vserver-LB-1, and then click Open.
3. On the Method and Persistence tab, under LB Method, select Source IP Destination IP Hash.
4. On the Advanced tab, under Redirection Mode, click MAC Based.
5. Select the Sessionless check box, and then click OK.

To set a service to use source IP address by using the command line interface

At the command prompt, type:

set service <ServiceName> -usip <Value>

Example

```
set service Service-ANY-1 -usip yes
```

To set a service to use source IP address by using the configuration utility

1. Navigate to Traffic Management > Load Balancing > Services.
2. On the Services pane, select the service, Service-ANY-1, and then click Open.
3. On the Advanced tab, under Settings, select the Use Source IP check box.
4. Click OK.
5. Repeat steps 1-5 for the services Service-ANY-2 and Service-ANY-3.

For USIP to function correctly, you must set it globally. For more information about configuring USIP globally, see "IP Addressing."

Use Case 10: Isolating Network Traffic using Listen Policies

Jun 08, 2015

A very common security requirement in a data center is to maintain network path isolation between the traffic of various applications or tenants. One application or tenant's traffic must be isolated from the traffic of other applications or tenants. For example, a financial services company would want to keep the traffic of its insurance department's applications separate from that of its financial services applications. In the past, this was easily achieved through physical separation of network service devices such as firewalls, load balancers, and IDP, and network monitoring and logical separation in the switching fabric.

As data center architectures evolve toward multi-tenant virtualized data centers, networking services in the aggregation layer of a data center are getting consolidated. This development has made network path isolation a critical component for network service devices and is driving the requirement for ADCs to be able to isolate traffic at the L4 to L7 levels. Furthermore, all the traffic of a particular tenant must go through a firewall before reaching the service layer.

To address the requirement of isolating the network paths, a NetScaler appliance identifies network domains and controls the traffic across the domains. The NetScaler solution has two main components: listen policies and shadow virtual servers.

Each network path to be isolated is assigned a virtual server on which a listen policy is defined so that the virtual server listens to traffic only from a specified network domain.

To isolate the traffic, listen policies can be based on a number of client parameters or their combinations, and the policies can be assigned priorities. The following table lists the parameters that can be used in listen policies for identifying the traffic.

Table 1. Client Parameters Used to Define Listen Policies

Category	Parameters
Ethernet protocol	Source MAC address, destination MAC address
Network interface	Network ID, receiving throughput, sending throughput, transmission throughput
IP protocol	Source IP address, destination IP address
IPv6 protocol	Source IPv6 address, destination IPv6 address
TCP protocol	Source port, destination port, maximum segment size, payload, and other options
UDP protocol	Source port, destination port
VLAN	ID

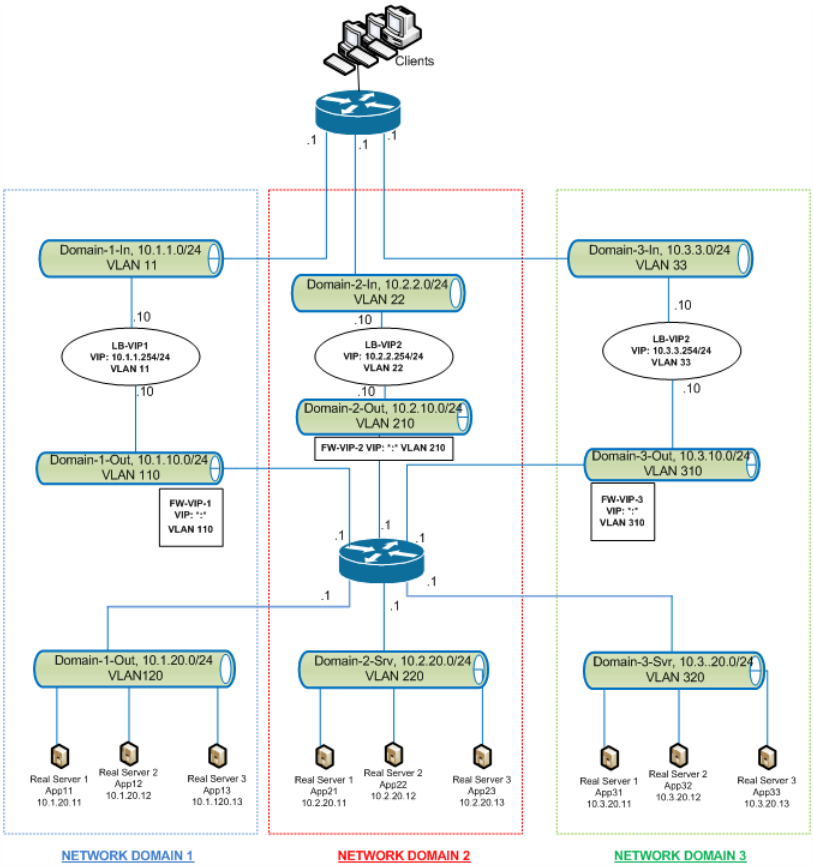
On the NetScaler appliance, a virtual server is configured for each domain, with a listen policy specifying that the virtual server is to listen only to traffic for that domain. Also configured for each domain is a shadow load balancing virtual server, which listens to traffic destined for any domain. Each of the shadow load balancing virtual servers has a wildcard (*) IP address and port, and its service type is set to ANY.

In each domain, a firewall for the domain is bound as a service to the shadow load balancing virtual server, which forwards all traffic through the firewall. Local traffic is forwarded to its destination, and traffic destined for another domain is forwarded to the firewall for that domain. The shadow load balancing virtual servers are configured for MAC mode redirection.

How Network Paths Are Isolated

The following figure shows a typical traffic flow across domains. Consider the traffic flow within Network Domain 1, and between Network Domain 1 and Network Domain 2.

Figure 1. Isolating Network Path



Traffic within Network Domain 1

Network Domain 1 has three VLANs: VLAN 11, VLAN110, and VLAN120. The following steps describe the traffic flow.

- A client from VLAN 11 sends a request for a service available from the service pool in VLAN 120.
- The load balancing virtual server LB-VIP1, which is configured to listen to traffic from VLAN 11, receives the request and forwards the request to VLAN 110. The virtual server in VLAN 110 forwards the request to shadow load balancing virtual server FW-VIP-1.
- FW-VIP-1, which is configured to listen to traffic from VLAN 110, receives the request and forwards it to VLAN 120.
- The load balancing virtual server in VLAN 120 load balances the request to one of the physical servers, App11, App12, or App13.
- The response sent by the physical server returns by the same path to the client in VLAN 11.

This configuration ensures that traffic is always segregated inside the NetScaler for all the traffic that originates from a client.

Traffic between Network Domain 1 and Network Domain 2

Network Domain 1 has three VLANs: VLAN 11, VLAN 110, and VLAN 120. Network Domain 2 also has three VLANS: VLAN 22, VLAN 210, and VLAN 220. The following steps describe the traffic flow from VALN 11 to VLAN 22.

- A client from VLAN 11, which belongs to Network Domain 1, sends a request for a service available from the service pool in VLAN 220, which belongs to the Network Domain 2.
- In Network Domain 1, the load balancing virtual server LB-VIP1, which is configured to listen to traffic from VLAN 11, receives the request and forwards the request to VLAN 110.
- Shadow load balancing virtual server FW-VIP-1, which is configured to listen to VLAN 110 traffic destined to any other domain, receives the request and forwards it to firewall virtual server FW-VIP-2 because the request is destined to a physical server in Network Domain 2.
- In Network Domain 2, FW-VIP-2 forwards the request to VLAN 220.
- The load balancing virtual server in VLAN 220 load balances the request to one of the physical servers, App21, App22, or App23.
- The response sent by the physical server returns by the same path through the firewall in Network Domain 2 and then to Network Domain 1 to reach the client in VLAN 11.

Configuration Steps

To configure network path isolation by using listen policies, do the following:

- Add listen policy expressions. Each expression specifies a domain to which traffic is destined. You can use the VLAN ID or other parameters to identify the traffic. For more details, see "Client Parameters Used to Define Listen Policies."
- For each network domain, configure two virtual servers as follows:

- Create a load balancing virtual server for which you specify a listen policy that identifies the traffic destined for this domain. You can specify the name of an expression created earlier, or you can create a new expression while creating the virtual server.
- Create another load balancing virtual server, referred to as shadow virtual server, for which you specify a listen policy expression that applies to traffic destined for any domain. On this virtual server, set the service type to ANY and the IP address and port to an asterisk (*). Enable MAC-based forwarding on this virtual server.
- Enable the L2 Connection option on both the virtual servers.
  Generally, to identify a connection, the NetScaler uses the 4-tuple of client IP address, client port, destination IP address, and destination port. When you enable the L2 Connection option, the Layer 2 parameters of the connection (channel number, MAC address, and VLAN ID) are used in addition to the normal 4-tuple.

- Add services representing the server pools in the domain, and bind them to the virtual server.
- Configure the firewall for each domain as a service, and bind all of the firewall services to the shadow virtual server.

To isolate network traffic by using the command line interface

At the command prompt, type the following commands:

- add policy expression <expressionName> <listenPolicyExpression>
- add lb vserver <name> <serviceType> <ip> <port> -l2conn ON -listenPolicy <expressionName>
  Add a load balancing virtual server for each domain. This virtual server is for traffic of the same domain.
- add lb vserver <name> ANY * * -l2conn ON -m MAC -listenPolicy <expressionName>
  Add a shadow load balancing virtual server for each domain. This virtual server is for traffic of other domains.

Example

```
add policy expression e110 client.vlan.id==110
add policy expression e210 client.vlan.id==210
add policy expression e310 client.vlan.id==310
add policy expression e11 client.vlan.id==11
add policy expression e22 client.vlan.id==22
add policy expression e33 client.vlan.id==33

add lb vserver LB-VIP1 HTTP 10.1.1.254 80 -persistenceType NONE -listenPolicy e11
-cltTimeout 180 -l2Conn ON

add lb vserver LB-VIP2 HTTP 10.2.2.254 80 -persistenceType NONE - listenPolicy e22
-cltTimeout 180 -l2Conn ON

add lb vserver LB-VIP3 HTTP 10.3.3.254 80 -persistenceType NONE - listenPolicy e33
-cltTimeout 180 -l2Conn ON


add lb vserver FW-VIP-1 ANY * * -persistenceType NONE -lbMethod ROUNDROBIN - listenPolicy e110 -Listenpriority 1 -m MAC -cltTimeout 120

add lb vserver FW-VIP-2 ANY * * -persistenceType NONE -lbMethod ROUNDROBIN - listenPolicy e210 -Listenpriority 2 -m MAC -cltTimeout 120

add lb vserver FW-VIP-3 ANY * * -persistenceType NONE -lbMethod ROUNDROBIN - listenPolicy e310 -Listenpriority 3 -m MAC -cltTimeout 120


add service RD-1 10.1.1.1 ANY * -gslb NONE -maxClient 0 -maxReq 0 -cip DISABLED
-usip NO -useproxyport NO -sp ON -cltTimeout 120 -svrTimeout 120 -CKA NO -TCPB NO -CMP NO

add service RD-2 10.2.2.1 ANY * -gslb NONE -maxClient 0 -maxReq 0 -cip DISABLED
-usip NO -useproxyport NO -sp ON -cltTimeout 120 -svrTimeout 120 -CKA NO -TCPB NO -CMP NO

add service RD-3 10.3.3.1 ANY * -gslb NONE -maxClient 0 -maxReq 0 -cip DISABLED
-usip NO -useproxyport NO -sp ON -cltTimeout 120 -svrTimeout 120 -CKA NO -TCPB NO -CMP NO


bind lb vserver FW-VIP-1 RD-1

bind lb vserver FW-VIP-2 RD-2
```

bind lb vserver FW-VIP-3 RD-3

To isolate network traffic by using the configuration utility

1. Add services representing the servers, as described in "Creating a Service."
2. Add each firewall as a service:
   1. Navigate to Traffic Management > Load Balancing > Services
   2. In the details pane, click Add.
   3. In the Create Service dialog box, specify values for the following parameters:

      - Service Name*—The name that you assign to the service.
      - Protocol*—Select ANY from the drop-down list.
      - Server*—The firewall's IP address.
      - Port*—Specify a value of 80.

      *A required parameter

   4. Click Create.
   5. From the Services pane, open the services you created and verify the settings.
3. Configure a load balancing virtual server.
   1. In the navigation pane, expand Load Balancing, and then click Virtual Servers.
   2. In the details pane, click Add.
   3. In the Create Virtual Server (Load Balancing) dialog box, specify values for the following parameters, which are described in "Creating a Virtual Server."

      - Name*
      - Protocol*
      - IP Address*
      - Port*

      *A required parameter

   4. On the Services tab, select the corresponding services.
   5. On the Advanced tab, select the L2 Connection check box and, for Redirection Mode, select MAC Based. Then, click the Listen Policy link and create the listen policy for the virtual server.
   6. Click Create.
4. Configure the shadow load balancing virtual server.
   1. For the shadow virtual server, specify

      - Protocol—ANY
      - IP Address*—*
      - Port*—*

      *A required parameter

   2. Bind the firewall services to the shadow virtual server.
5. For each network domain, repeat steps 3 and 4.
6. From the Load Balancing Virtual Servers pane, open the virtual servers that you created and verify the settings.

Use Case 11: Configuring XenDesktop for Load Balancing

Jun 08, 2015

For an improved performance in the delivery of virtual desktop applications, you can integrate the NetScaler appliance with Citrix XenDesktop and use the NetScaler load balancing feature to distribute the load across the Web Interface servers and the Desktop Delivery Controller (DDC) servers.

Generally, you use XenDesktop in situations where applications are not compatible with running on a terminal server or XenApp, or if each virtual desktop has unique requirements. In such cases, you need one desktop host for each user that connects. However, the hosts can be pooled so that you need only one host for each currently connected user.

The core application service deployed for XenDesktop is the Desktop Delivery Controller (DDC). The DDC is installed on a server, and its main function is to register desktop hosts and broker client connections to them.

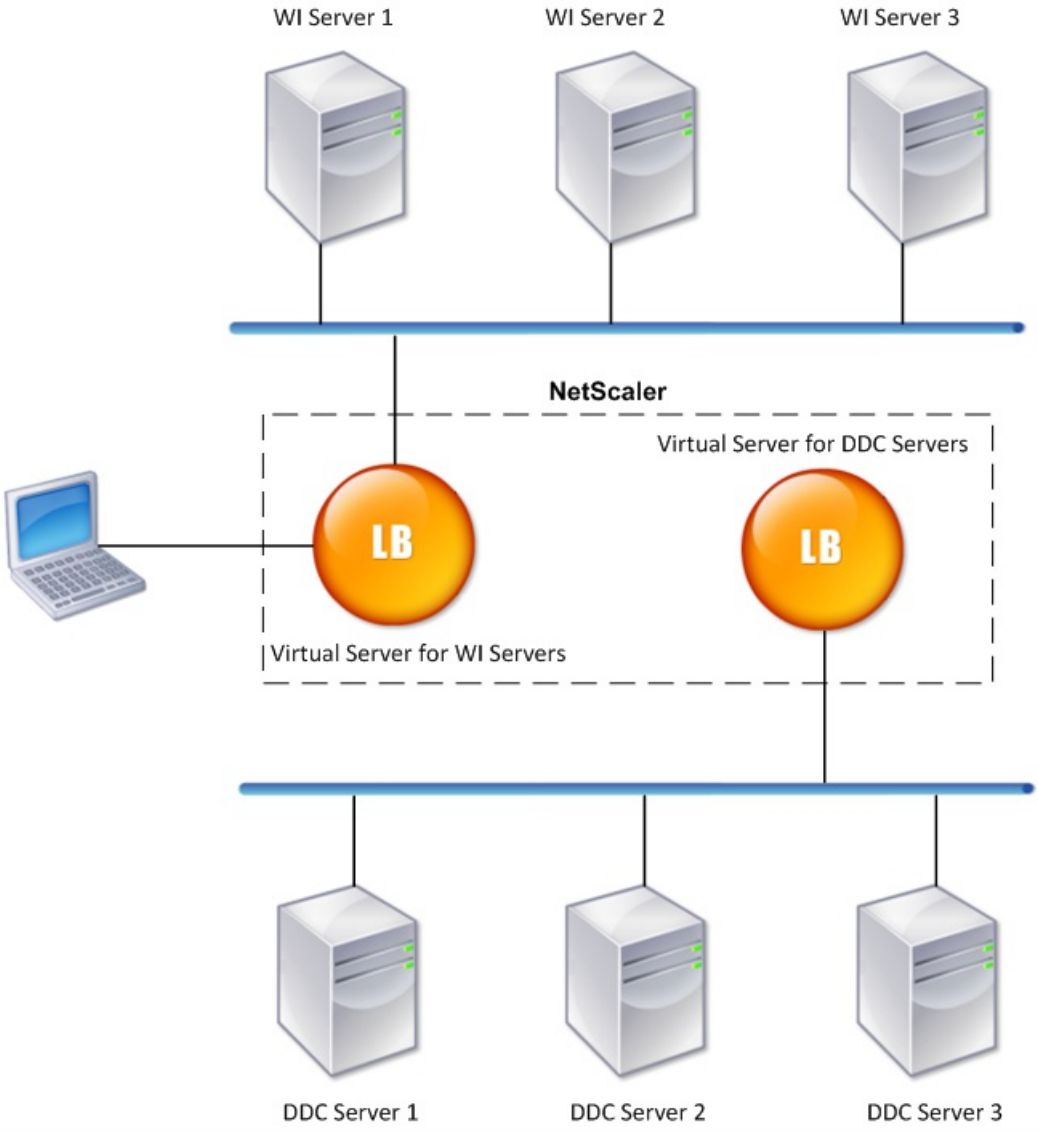
The DDC also authenticates users and manages the assembly of the users' virtual desktop environments by controlling the state of the desktops, and starting and stopping the desktops.

Generally, multiple DDCs are installed to enhance availability.

The Web Interface servers provide secure access to virtual desktops. The Web Interface is the initial connection portal to the Desktop Delivery Controller (DDC). The Web browser on the user's device sends information to the Web server, which communicates with the server farm to provide the user with access to the virtual desktop.

The following figure shows the topology of a NetScaler appliance working with XenDesktop.

Figure 1. **Load Balancing of XenDesktop**



Note: Although you can use the HTTP protocol, Citrix recommends that you use SSL for communication between the client and the NetScaler. You can use the HTTP protocol for communication between the NetScaler and the DDC servers even though you use the SSL protocol for communication with the client.
A wizard is available for configuring basic load balancing in a XenDesktop deployment. You can use the wizard to configure Web interface servers and a virtual server for them, and DDC servers and a virtual server for them. The virtual servers that you configure are bound to services specified as Web Interface services and DDC services. Each virtual server is configured with the default load balancing method, and the default features are enabled. A monitor is created and bound to each virtual server.

The wizard creates a basic setup, with default values for options such as the load balancing method, policies, persistence, and advanced settings. You can change any of the values if necessary.

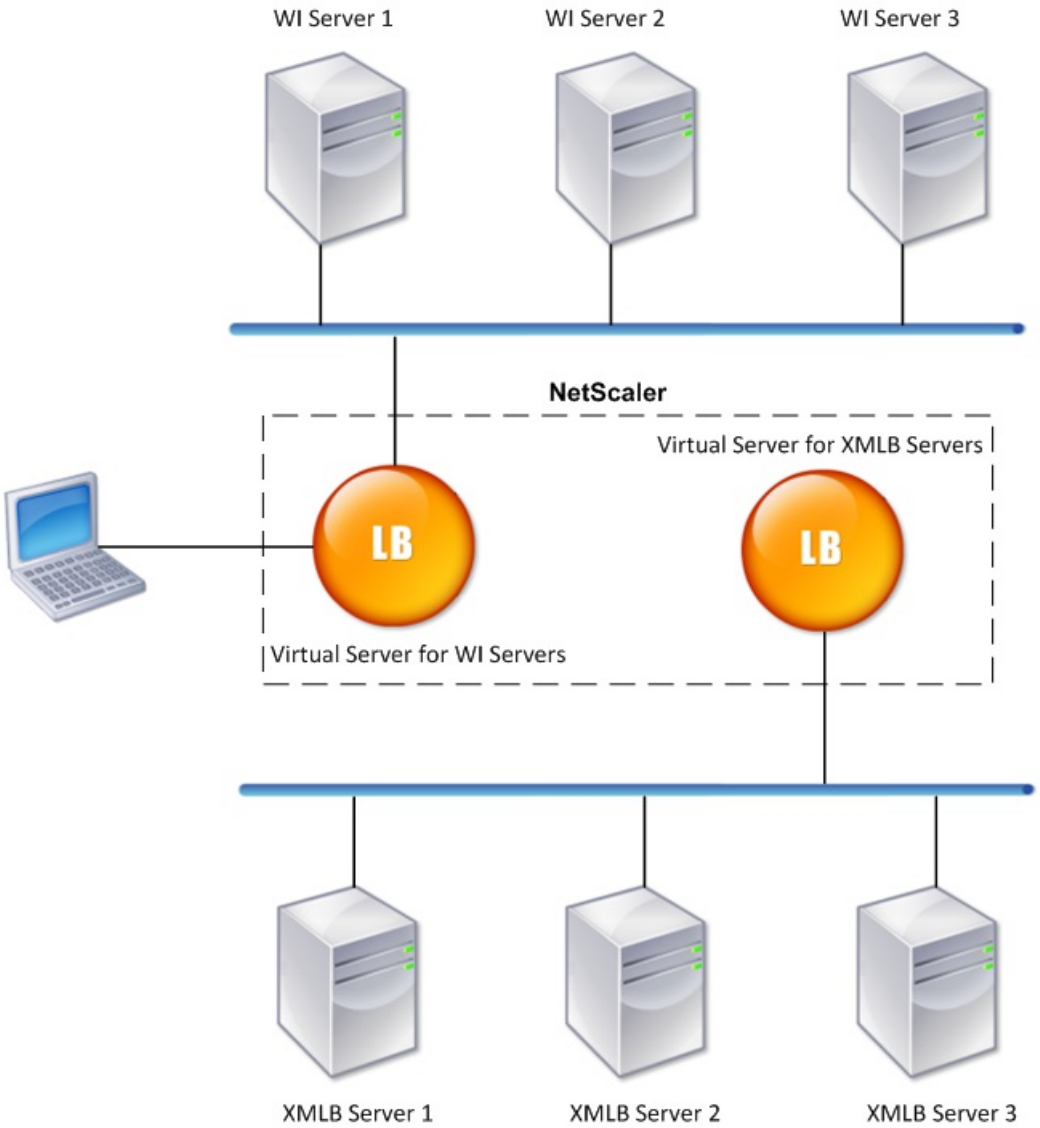
1. Navigate to Traffic Management > Load Balancing.
2. In the Getting Started group, click Load balancing wizard for Citrix XenDesktop.
3. Follow the instructions presented by the wizard.

Use Case 12: Configuring XenApp for Load Balancing

Jun 08, 2015

For efficient delivery of applications, you can integrate the NetScaler appliance with Citrix XenApp and use the NetScaler load balancing feature to distribute the load across the XenApp server farms. The following figure is a topology diagram of such a setup.

Figure 1. Load Balancing of XenApp



The Web Interface servers provide secure access to XenApp application resources through the user's Web browser. The Web Interface client presents to the users all the resources, such as applications, content, and desktops that are made available in the XenApp server farms. Users can access the published resources through a standard Web browser or through the Citrix online plug-in.

The Web browser on the user's device sends information to the Web server, which communicates with the servers on the server farm to provide the user with access to the resources.

The Web Interface and the XML Broker are complementary services. The Web Interface provides users with access to applications, and the XML Broker evaluates the user's permissions to determine which applications appear in the Web

Interface.

The XML service is installed on all the servers in the server farm. The XML service specified in the Web Interface functions as an XML broker. On the basis of the user credentials passed by the Web Interface server, the XML Broker server sends a list of applications accessible to the user.

In large enterprises where multiple Web Interface servers and XML Broker servers are deployed, Citrix recommends load balancing these servers by using NetScaler. Configure one virtual server to load balance all of the Web Interface servers and another for all of the XML Broker servers. The load balancing method and other features can be configured on the virtual server as required.

Note: Although you can use the HTTP protocol, Citrix recommends that you use SSL for communication between the client and the NetScaler. You can use the HTTP protocol for communication between the NetScaler and the WI servers even though you use the SSL protocol for communication with the client.
The configuration utility provides a wizard for setting up basic load balancing for XenApp.

Through this wizard, you can configure Web Interface servers and a virtual server for them, and XML Broker servers and a virtual server for them. You can also specify the site through which the status of Web Interface servers can be monitored and the software application used to monitor the status of the XML Broker servers.

When you complete the wizard, a basic load balancing setup is configured on the NetScaler. The specified virtual servers are created and bound to the services specified as Web Interface services and XML Broker services. Each virtual server is configured with the default load balancing method, and the default features are enabled. A monitor is created and bound to each virtual server.

The wizard creates a basic setup with default values for options such as the load balancing method, policies, persistence, and advanced settings. You can change any of the values if necessary.

1. Navigate to Traffic Management > Load Balancing.
2. In the Getting Started group, click Load balancing wizard for Citrix XenApp.
3. Follow the instructions presented by the wizard.

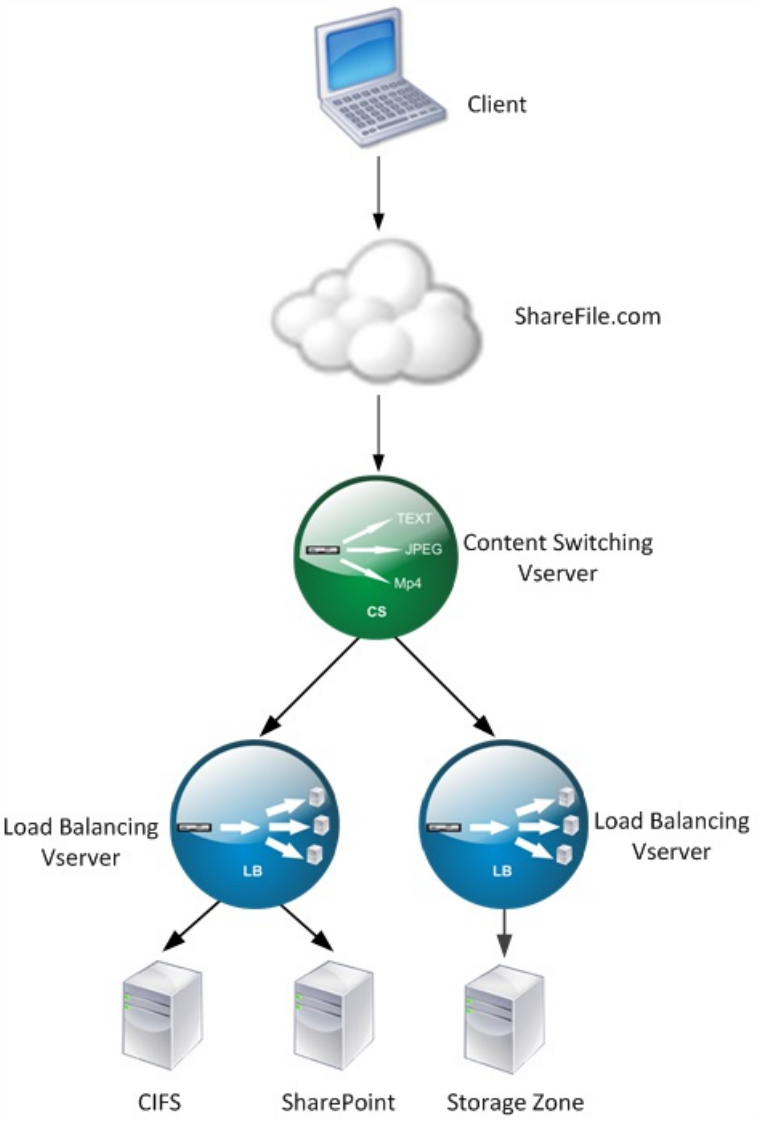
Use Case 13: ShareFile Wizard for Load Balancing Citrix ShareFile

Jun 08, 2015

You can configure load balancing for Citrix ShareFile using the wizard. The Citrix ShareFile wizard helps in setting up load balancing configuration for ShareFile site based on the type of content requested. The content switching server directs the request based on whether it is a StorageZone, CIFS or a SharePoint request. The content switching is based on policies. The wizard auto generates the policies to identify whether the request is for StorageZone, CIFS or SharePoint. The content switching virtual server uses these policies to direct the request to the correct load balancing server.

A typical data flow can be depicted as shown in the diagram below.

Figure 1. ShareFile Data Load Balancing



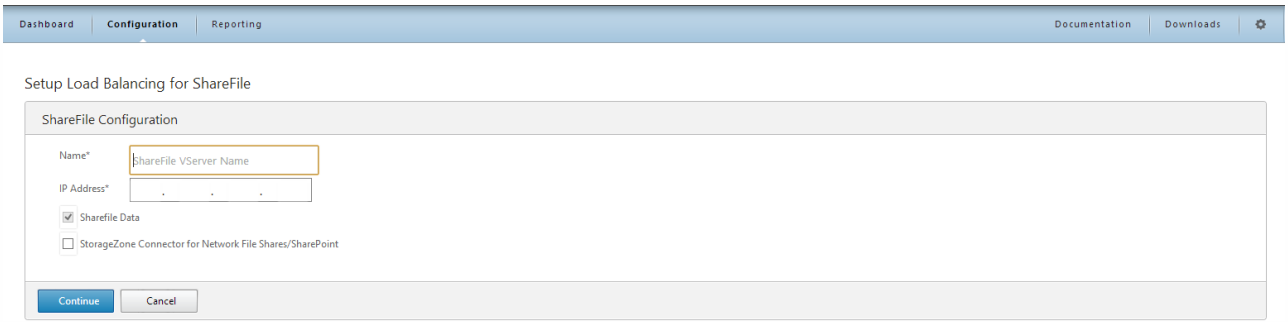
You can view the load balancing virtual servers that the ShareFile wizard creates by navigating to Traffic Management >Virtual Servers and Services > Virtual Servers. You cannot manually remove the virtual servers created using the ShareFile

wizard. Use the wizard to remove the virtual servers.

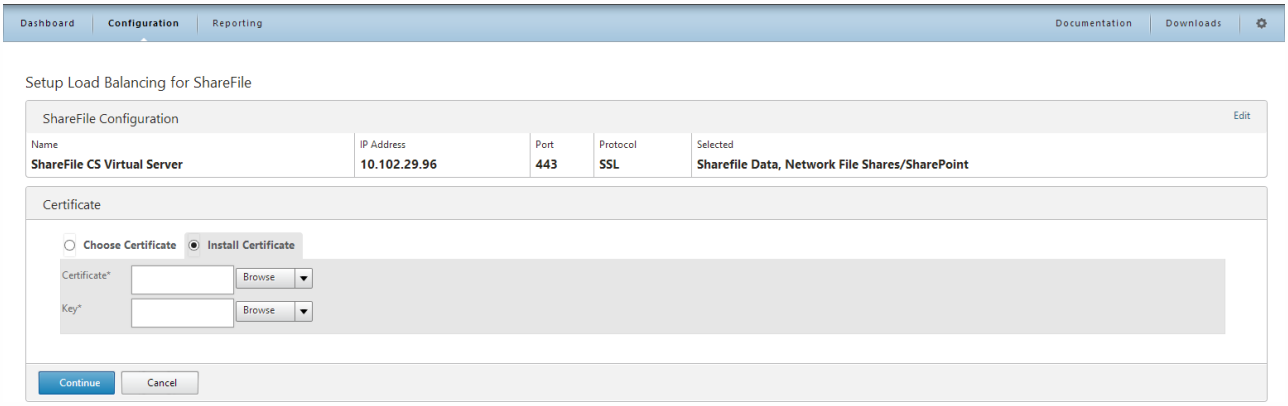
NetScaler uses the LDAP authentication for SharePoint or CIFS request. Hash authentication is used for authenticating requests for StorageZones.

Updated: 2015-06-04

1. In the navigation pane, click Load Balancing.
2. Navigate to Traffic Management > Load Balancing.
3. Under Citrix ShareFile, click Setup NetScaler for ShareFile.
4. On the Setup Load Balancing for ShareFile page, provide the following information:
   - Name: Name of the content switching virtual server.
   - IP Address: IP address of the content switching virtual server.
   - If you want to setup load balancing for CIFS or SharePoint, click the StorageZone Connector for Network File Shares/SharePoint checkbox and then click Continue. By default ShareFile Data checkbox is selected.

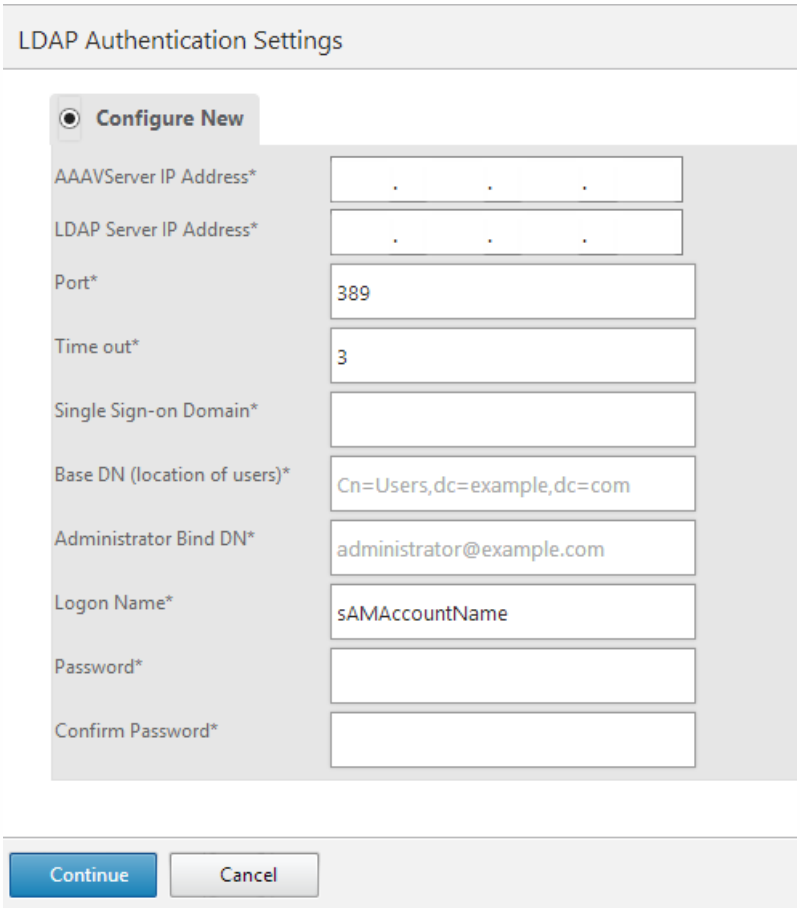


5. Provide a valid certificate. If you have a certificate, click Choose Certificate and from the drop-down list select the certificate. If you have to install a certificate, click Install Certificate and provide the Certificate-Key pair.



6. Click Continue.
7. In the Add New StorageZone Controller dialog box, specify the values of the following parameters:
   - StorageZone Controller IP Address— IP address
   - Port— Port number. The default value is 443.
   - Protocol— Select from HTTPS or HTTP

8. Click Create and then click Done. The wizard automatically creates a service and autogenerate the name of the service.
9. If you chose load balancing for CIFS or SharePoint in step 4.c, then specify the values for LDAP Authentication Settings:
   - AAAVServer IP Address— IP address of AAA virtual server
   - LDAP Server IP Address— IP Address of the LDAP server
   - Port— Port number. The default value is 389
   - Time out— The time out value in minutes
   - Single Sign-on Domain— Single sign-on domain name
   - Base DN— Base domain name
   - Administrator Bind DN— LDAP account name with the domain name, for example, adminstrator@domainname.com
   - Logon Name— Logon name is the sAMAccount name
   - Password and Confirm Password— Enter the password and confirm the password



10. Click Continue and then click Done.

To remove load balancing configuration for ShareFile

1. Click on Configuration > Traffic Management > Load Balancing.

2. On the Load Balancing page, under Citrix ShareFile click on Remove ShareFile Configuration.

Troubleshooting

Aug 01, 2013

If the load balancing does not work as expected after you have configured it, you can use some common tools to access NetScaler resources and diagnose the problem.

Updated: 2013-08-01

For best results, use the following resources to troubleshoot a content switching issue on a NetScaler appliance:

- Latest ns.conf file
- Relevant newnslog files
- Ethereal packet traces recorded on the appliance and relevant client, if possible
- The ns.log file

In addition to the above resources, the following tools expedite troubleshooting:

- A browser add-on tool that can display HTTP headers. This can be used to troubleshoot persistency related issues.
- The Wireshark application customized for the NetScaler trace files.

Updated: 2015-06-11

- Issue
  I created a user script for monitoring, but it is not working.

  Resolution

  Check the number of arguments in the script. The limit is 512. A script with more than 512 arguments might not work properly. Use the nsumon-debug.pl script from the NetScaler command line to debug the script.

- Issue
  I see a lot of monitor probes, and they seem to be increasing the network traffic unnecessarily. Is there a way to turn off the monitor probes?

  Resolution

  You can turn off the monitor probe connections, by disabling the monitor or setting the value of the healthMonitor parameter in the set service command to NO. With the NO option, the appliance shows the service as UP at all times.

- Issue
  I have set up monitors for services, but connections are still directed to servers that are DOWN.

  Resolution

  You probably need to decrease the monitor probe intervals. The NetScaler appliance does not detect the DOWN state until the monitor sends a probe.

- Issue
  A metric bound to the monitor is present in the local and custom metric tables.

Resolution

Add the local prefix to the metric name if the metric is chosen from the local metric table. However, if the metric is chosen from the custom table, you don't need to add any prefix.

- Issue
  The monitor probes to a service are not reaching the service.

  Resolution

  Check whether you have set a limit on the number of connections for a service. If yes, exempt monitor-probe connections from this limit by setting the monitorSkipMaxClient parameter to ENABLED.

- Issue
  I am able to ping the servers, but the state of the services is always shown as DOWN.

  Resolution

  Check the type of monitors configured. For example, if a server is not configured for SSL and you use an HTTPS monitor, the state of the service is marked as DOWN. In this case using a TCP monitor should change the state of the service to UP.

- Issue
  Setting a weight for load monitors does not help in deciding the state of the service.

  Resolution

  Load monitors cannot decide the state of the service. Therefore, setting a weight on the load monitors is inappropriate.

- Issue
  A service is not stable.

  Resolution

  Consider troubleshooting the following components:
  - Verify that a correct server is bound to the service.
  - Verify the type of monitor bound to the service.
  - Verify the reasons for the monitor failures. You can open service from the Services page and verify the details for the number of probes, failures, and last response status for the monitor in the Monitors tab of the Configure service dialog box. To display the details, click the monitor configured.
  - If it is a custom monitor, bind a TCP or ping monitor to the service and verify the status of the monitor. If this resolves the issue, there is some problem with the custom monitor and the monitor requires further investigation.
  - You can record packet traces on the NetScaler appliance and verify the monitor probes and server response for further investigation.

- Issue
  The virtual IP (VIP) address is not stable or its status is displayed as DOWN.

  Resolution

  Consider troubleshooting the following components:
  - Verify that the load balancing feature is licensed.

- Verify that the feature is enabled.
- Verify that an appropriate service is bound to the virtual server.
- If the status of the VIP address is displayed as DOWN, verify that an administrator has enabled the service. If it is not, the status of the service should be Out-Of-Service. In such as case, you must enable the service and verify if the issue is resolved.
- Verify the service(s) bound to the virtual server and complete the troubleshooting steps mentioned for service not stable issue.
- If the VIP address is not stable, all the services bound to the virtual server should fail. Therefore, verify if all the services are failing at the same time. If it is so, there is a network issue between the NetScaler appliance and the servers.

- Issue
  The site is experiencing uneven load balancing.

  Resolution

  Consider troubleshooting the following components:
  - Verify the load balancing method configured on the appliance.
  - Verify weights associated with the services are as expected.
  - If the load balancing method is other than round robin, verify the number of connections to the server logged in the newnslog file. You can run the following command to verify the number on the newnslog file:
    # nsconmsg –K <newnslog_file> -s ConLb=2 –d oldconmsg

    Verify the services for the specific virtual server and check for the Response time, Open Established connections (OE), Hits, Persistent Hits and persistent rate (P) to troubleshoot the issue further.

  - If the load balancing method is round robin, verify the persistent Hits as mentioned in the preceding step. Additionally, verify if the service is not stable. If it is not, complete the troubleshooting steps mentioned for service not stable issue
  - Verify if persistency is configured on the appliance.
  - Verify is any service is not stable. If yes, complete the troubleshooting steps mentioned for service not stable issue.

- Issue
  The service status is displayed as DOWN.

  Resolution

  Consider troubleshooting the following components:
  - Verify whether a SNIP or MIP address is configured.
  - Verify that appropriate monitors are bound to the service.
  - If custom monitors are bound to the service, bind a TCP or ping monitor to the service and verify the status of the monitor. If this resolves the issue, there is some problem with the custom monitor and the monitor requires further investigation.
  - Verify if the status of service is displayed as DOWN for the server that is in another subnet. If yes, verify if Use Subnet IP (USNIP) resolves the issue because this could be due to the MIP address being unable to communicate to the server.

- Issue
  There is an issue with the response time.

  Resolution

Consider troubleshooting the following components:

- Verify the server response time from the service stats either by running the following command:
  # nsconmsg –K <newnslog_file> -s ConLb=2 –d oldconmsg
- Check for service not stable and service status being displayed as DOWN issues.

- Issue
  One of the servers is serving more requests than the other load balanced servers.

  Resolution

  Consider troubleshooting the following components:

  - Verify the load balancing method. Use the round robin method to distribute the client request equally regardless of the load on the servers.
  - Determine whether persistence is enabled for the load balancing configuration. If persistence is enabled, a given servers might be carrying a heavier load to maintain its session, especially If the persistence sessions are long.
  - Verify whether weights are assigned to each service. Assigning proper weights helps in proper load distribution.

- Issue
  Connections to a specific load balanced server are stalled. For example, all connections to one Outlook server might be stalled.

  Resolution

  Consider troubleshooting the following components:
  - Verify the load balance method. If it is round robin, consider changing the method to least connections.
  - Consider reducing the monitor time-out period. A shorter timeout period helps in marking a service as DOWN sooner, which would help in directing the traffic to server which is functional.
  - If the connections are stalled for a long period, surge-queue might build. Consider flushing the surge-queue to avoid a sudden spike in load on the server.
  - If the servers are working at their maximum level, consider adding a new server for better performance.

- Issue
  A majority of the connections are directed to a particular server, even when the least connections method for load balancing is configured.

  Resolution

  Determine whether persistence is configured and is of type source IP. If source IP persistence is configured even with the least connections method, the requests go to a specific server. The server's IP address is required for maintaining the session information. Consider using HTTP Cookies based persistence.

- Troubleshooting Tips
  For other issues, consider following tips to troubleshoot an issue not listed above:
  - If multiple load monitors are bound to a service, the load on the service is the sum of all the values on the load monitors bound to it. For load balancing to work properly, you must bind the same set of monitors to all the services.
  - If you disable a load monitor bound to the service and the service is bound to a virtual server, the virtual server uses the round robin method for load balancing.
  - When you bind a service to a virtual server where the load balancing method is CUSTOMLOAD and the service status is UP, the virtual server uses the initial round robin method for load balancing. It continues to be in round robin if the

service has no custom load monitors, or if status of at least one of the custom load monitors is not UP.

- All the services that are bound to a virtual server where the load balancing method is CUSTOMLOAD, the services must have load monitors bound to them.
- The CUSTOMLOAD load balancing method also follows startup round robin.
- If you disable a metric-based binding and this is the last active metric, the specific virtual server uses the round robin method for load balancing. A metric is disabled by setting the metric threshold to zero.
- When a metric bound to a monitor crosses the threshold value, that particular service is not considered for load balancing. If all the services have reached the threshold, the virtual server uses the round robin method for load balancing and an error message “5xx - server busy error” is displayed.
- A maximum of 10 metrics from a custom table can be bound to the monitor.
- The OIDs must be scalar variables.
- For successful load balancing, the interval must be as low as possible. If the interval is high, the time period for retrieving the load value increases. As a result, load balancing takes place using improper values.
- A user cannot modify the local table.

SSL Offload and Acceleration

Dec 08, 2014

A Citrix® NetScaler® appliance configured for SSL acceleration transparently accelerates SSL transactions by offloading SSL processing from the server. To configure SSL offloading, you configure a virtual server to intercept and process SSL transactions, and send the decrypted traffic to the server (unless you configure end-to-end encryption, in which case the traffic is re-encrypted). Upon receiving the response from the server, the appliance completes the secure transaction with the client. From the client's perspective, the transaction seems to be directly with the server. A NetScaler configured for SSL acceleration also performs other configured functions, such as load balancing.

Configuring SSL offloading requires an SSL certificate and key pair, which you must obtain if you do not already have an SSL certificate. Other SSL-related tasks that you might need to perform include managing certificates, managing certificate revocation lists, configuring client authentication, and managing SSL actions and policies.

A non-FIPS NetScaler appliance stores the server's private key on the hard disk. On a FIPS appliance, the key is stored in a cryptographic module known as a hardware security module (HSM). Only the MPX 9700/10500/12500/15500 appliances support a FIPS card, so other NetScaler models cannot be equipped with an HSM.

Beginning with release 10.5, build 52.1115.e, all NetScaler appliances that do not support a FIPS card (including virtual appliances) support the Thales nShield® Connect external HSM. (MPX 9700/10500/12500/15500 appliances do not support an external HSM.)

Note: FIPS-related options for some of the SSL configuration procedures described in this document are specific to a FIPS-enabled NetScaler.

Configuring SSL Offloading

Jun 03, 2015

To configure SSL offloading, you must enable SSL processing on the NetScaler appliance and configure an SSL based virtual server that will intercept SSL traffic, decrypt the traffic, and forward it to a service that is bound to the virtual server. To enable SSL offloading, you must import a valid certificate and key and bind the pair to the virtual server.

To configure SSL offloading, see the following sections:

- Enabling SSL Processing
- Configuring Services
- Configuring an SSL-Based Virtual Server
- Binding Services to the SSL-Based Virtual Server
- Adding or Updating a Certificate-Key Pair
- Binding the Certificate-Key Pair to the SSL-Based Virtual Server
- Configuring an SSL Virtual Server for Secure Hosting of Multiple Sites
- Importing SSL Files from Remote Hosts

Enabling SSL Processing

Nov 11, 2013

To process SSL traffic, you must enable SSL processing. You can configure SSL based entities, such as virtual servers and services, without enabling SSL processing, but they will not work until SSL processing is enabled.

At the command prompt, type:

- enable ns feature ssl
- show ns feature

Example

```
> enable ns feature SSL
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection          SP              ON
 3)    Load Balancing           LB              ON
 .
 .
 .
 9)    SSL Offloading           SSL             ON
 .
 .
 .
 24)   NetScaler Push           push            OFF
Done
```

1. In the navigation pane, expand System, and then click Settings.
2. Under Modes and Features, click Configure basic features.
3. Select the SSL Offloading check box, and then click OK.
4. In the Enable/Disable Feature(s)? message box, click Yes. A message appears in the status bar, stating that the feature has been enabled.

Configuring Services

Aug 20, 2013

On the NetScaler appliance, a service represents a physical server or an application on a physical server. Once configured, services are in the disabled state until the appliance can reach the physical server on the network and monitor its status.

At the command prompt, type the following commands to add a service and verify the configuration:

- add service <name> (<IP> | <serverName>) <serviceType> <port>
- show service <serviceName>

Example

```
> add service ssl1 10.102.29.252 HTTP 80
 Done
> show service ssl1
      ssl1 (10.102.29.252:80) - HTTP
      State: UP
      Last state change was at Thu Nov 12 05:26:31 2009
      Time since last state change: 0 days, 00:00:06.750
      Server Name: 10.102.29.252
      Server ID : 0   Monitor Threshold : 0
      Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
      Use Source IP: NO
      Client Keepalive(CKA): NO
      Access Down Service: NO
      TCP Buffering(TCPB): YES
      HTTP Compression(CMP): YES
      Idle timeout: Client: 180 sec   Server: 360 sec
      Client IP: DISABLED
      Cacheable: NO
      SC: OFF
      SP: ON
      Down state flush: ENABLED

1)     Monitor Name: tcp-default
            State: UP       Weight: 1
            Probes: 2       Failed [Total: 0 Current: 0]
            Last response: Success - TCP syn+ack received.
            Response Time: N/A
 Done
```

To modify a service, use the set service command, which is just like using the add service command, except that you enter the name of an existing service. To remove a service, use the rm service command, which accepts only the <name> argument.

1. Navigate to Traffic Management > SSL Offload > Services.
2. In the Details pane, do one of the following:
   - To create a new service, click Add.
   - To modify an existing service, select the service, and then click Open.
3. In the Create Service or Configure Service dialog box, specify values for the following parameters:
   - Service Name*
   - Server*
   - Protocol*
   - Port*

   * A required parameter
4. Click Create or OK, and then click Close. In the Services pane, select the service that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configuring an SSL-Based Virtual Server

Sep 04, 2014

Secure sessions require establishing a connection between the client and an SSL-based virtual server on the NetScaler appliance. The SSL virtual server intercepts SSL traffic, decrypts it and processes it before sending it to services that are bound to the virtual server.

Note: The SSL virtual server is marked as down on the NetScaler appliance until a valid certificate / key pair and at least one service are bound to it. An SSL based virtual server is a load balancing virtual server of protocol type SSL or SSL_TCP. The load balancing feature must be enabled on the NetScaler.

At the command prompt, type the following commands to create an SSL-based virtual server and verify the configuration:

- add lb vserver <name> (serviceType) <IPAddress> <port>
- show lb vserver <name>

Example

```
> add lb vserver vssl  SSL 10.102.29.133 443
 Done
> show ssl vserver vssl

        Advanced SSL configuration for VServer vssl:
        DH: DISABLED
        Ephemeral RSA: ENABLED        Refresh Count: 0
        Session Reuse: ENABLED        Timeout: 120 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: DISABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)      Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
 Done
```

To modify the load balancing properties of an SSL virtual server, use the set lb vserver command, which is just like using the add lb vserver command, except that you enter the name of an existing vserver. To modify the SSL properties of an SSL-based virtual server, use the set ssl vserver command. For more information, see Customizing the SSL Configuration.

To remove an SSL virtual server, use the rm lb vserver command, which accepts only the <name> argument.

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.

2. In the Details pane, do one of the following:
   - To create a new virtual server, click Add.
   - To modify an existing virtual server, select the virtual server, and then click Open.
3. In the Create Virtual Server (SSL Offload) or Configure Virtual Server (SSL Offload) dialog box, specify values for the following parameters:
   - Name*
   - IP Address*
   - Protocol*
   - Port*

   * A required parameter
4. Click Create or OK, and then click Close. In the SSL Offload Virtual Servers pane, select the virtual server that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Binding Services to the SSL-Based Virtual Server

Aug 20, 2013

For the NetScaler appliance to forward decrypted SSL data to servers in the network, services representing these physical servers must be bound to the virtual server that receives the SSL data.

Because the link between the NetScaler and the physical server is typically secure, data transfer between the appliance and the physical server does not have to be encrypted. However, you can provide end-to end-encryption by encrypting data transfer between the NetScaler and the server. For details, see Configuring SSL Offloading with End-to-End Encryption.

Note: The Load Balancing feature should be enabled on the NetScaler appliance before you bind services to the SSL based virtual server.

At the command prompt, type the following commands to bind the service to the virtual server and verify the configuration:

- bind lb vserver <name> <serviceName>
- show lb vserver <name>

Example

```
> bind lb vserver vssl ssl1
 Done
> show lb vserver vssl
      vssl (10.102.29.133:443) - SSL  Type: ADDRESS
      State: DOWN[Certkey not bound]
      Last state change was at Thu Nov 12 05:31:17 2009 (+485 ms)
      Time since last state change: 0 days, 00:08:52.130
      Effective State: DOWN
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      No. of Bound Services :  1 (Total)      1 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
      Vserver IP and Port insertion: OFF
      Push: DISABLED  Push VServer:
      Push Multi Clients: NO
      Push Label Rule: none

1) ssl1 (10.102.29.252: 80) - HTTP State: UP    Weight: 1
 Done
```

At the command prompt, type the following command:

unbind lb vserver <name> <serviceName>

Example

```
unbind lb vserver vssl ssl1
```

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind the service, and click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, select the Services tab, and then select the check box in the Active column of the ssl service that you want to bind to the virtual server.
4. Click OK. A message appears in the status bar, stating that the service has been bound successfully

Adding or Updating a Certificate-Key Pair

Feb 16, 2016

For any SSL transaction, the server needs a valid certificate and the corresponding private and public key pair. The SSL data is encrypted with the server's public key, which is available through the server's certificate. Decryption requires the corresponding private key.

Because the NetScaler appliance offloads SSL transactions from the server, the server's certificate and private key must be present on the appliance, and the certificate must be paired with its corresponding private key. This certificate-key pair must then be bound to the virtual server that processes the SSL transactions.

Both the certificate and the key must be in local storage on the NetScaler appliance before they can be added to the appliance. If your certificate or key file is not on the appliance, upload it to the appliance before you create the pair.
Important: Certificates and keys are stored in the /nsconfig/ssl directory by default. If your certificates or keys are stored in any other location, you must provide the absolute path to the files on the NetScaler appliance. The NetScaler FIPS appliances do not support external keys (non-FIPS keys). On a FIPS appliance, you cannot load keys from a local storage device such as a hard disk or flash memory. The FIPS keys must be present in the Hardware Security Module (HSM) of the appliance.
On a NetScaler MPX appliance and a NetScaler FIPS appliance, only RSA private keys are supported. On a VPX virtual appliance, both RSA and DSA private keys are supported. On an SDX appliance if SSL chips are assigned to an instance, then only RSA private keys are supported. However, if SSL chips are not assigned to an instance, then both RSA and DSA private keys are supported. In all the cases, you can bind a CA certificate with either RSA or DSA keys.

Set the notification period and enable the expiry monitor to be prompted before the certificate expires.

Note: A certificate must be signed by using one of the following hash algorithms:
- MD5
- SHA-1
- SHA-224
- SHA-256
- SHA-384
- SHA-512

An MPX appliance supports certificates of 512 or more bits, up to the following sizes:
- 4096-bit server certificate on the virtual server
- 4096-bit client certificate on the service
- 4096-bit CA certificate (includes intermediate and root certificates)
- 4096-bit certificate on the back-end server
- 4096-bit client certificate (if client authentication is enabled on the virtual server)

A VPX virtual appliance supports certificates of 512 or more bits, up to the following sizes:
- 4096-bit server certificate on the virtual server
- 4096-bit client certificate on the service
- 4096-bit CA certificate (includes intermediate and root certificates)
- 2048-bit certificate on the back-end server
- 2048-bit client certificate (if client authentication is enabled on the virtual server)

> ### Note
>
> A NetScaler SDX appliance supports certificates of 512 or more bits. Each NetScaler VPX instance hosted on the appliance supports the certificate sizes listed above for a VPX virtual appliance. However, if an SSL chip is assigned to an instance, that instance supports the certificate sizes supported by an MPX appliance.

At the command prompt, type the following commands to add a certificate-key pair and verify the configuration:

- add ssl certKey <certkeyName> -cert <string>[(-key <string> [-password]) | -fipsKey <string>] [-inform (DER | PEM)] [<passplain>] [-expiryMonitor (ENABLED | DISABLED) [-notificationPeriod <positive_integer>]]
- show ssl certKey [<certkeyName>]

Example

```
> add ssl certKey sslckey -cert server_cert.pem -key server_key.pem -password ssl -expiryMonitor ENABLED -notificationPeriod 30
 Done
Note: For FIPS appliances, replace -key with -fipskey
> show ssl certKey sslckey
      Name: sslckey         Status: Valid,   Days to expiration:8418
      Version: 3
      Serial Number: 01
      Signature Algorithm: md5WithRSAEncryption
      Issuer:  C=US,ST=SJ,L=SJ,O=NS,OU=NSSSL,CN=www.root.com
      Validity
            Not Before: Jul 15 02:25:01 2005 GMT
            Not After : Nov 30 02:25:01 2032 GMT
      Subject:  C=US,ST=SJ,L=SJ,O=NS,OU=NSSSL,CN=www.server.com
      Public Key Algorithm: rsaEncryption
      Public Key size: 2048
 Done
```

To modify the expiry monitor or notification period in a certificate-key pair, use the set ssl certkey command. To replace the certificate or key in a certificate-key pair, use the update ssl certkey command. The update ssl certkey command has an additional parameter for overriding the domain check. For both commands, enter the name of an existing certificate-key pair. To remove an SSL certificate-key pair, use the rm ssl certkey command, which accepts only the <certkeyName> argument.

1. Navigate to Traffic Management > SSL > Certificates.
2. In the Details pane, do one of the following:
   - To add a new certificate-key pair, click Install.
   - To update an existing certificate-key pair, select a certificate and then, from the Action list, select Update.
3. In the Install Certificate or Update Certificate dialog box, set the following parameters:
   - Certificate-Key Pair Name*
   - Certificate File Name*
   - Private Key File Name
   - Password
   - Certificate Format
   - Notify When Expires
   - Notification Period
   - No Domain Check (Available in the Update Certificate dialog box only)

   * A required parameter
4. Click Install or OK, and then click Close. In the SSL Certificates pane, select the certificate that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Binding the Certificate-Key Pair to the SSL-Based Virtual Server

Aug 20, 2013

An SSL certificate is an integral element of the SSL encryption and decryption process. The certificate is used during an SSL handshake to establish the identity of the SSL server.

The certificate being used for processing SSL transactions must be bound to the virtual server that receives the SSL data. If you have multiple virtual servers receiving SSL data, a valid certificate-key pair must be bound to each of them.

You can use a valid, existing SSL certificate that you have uploaded to the NetScaler appliance. As an alternative for testing purposes, you can create your own SSL certificate on the appliance. Intermediate certificates created by using a FIPS key on the NetScaler cannot be bound to an SSL virtual server.

As a part of the SSL handshake, in the certificate request message during client authentication, the server lists the distinguished names (DNs) of all the certificate authorities (CAs) bound to the server from which it will accept a client certificate. If you do not want the DN name of a specific CA certificate to be sent to the SSL client, set the **skipCA** flag. This setting indicates that the particular CA certificate's distinguished name should not be sent to the SSL client.

For details on how to create your own certificate, see Managing Certificates.

Note: Citrix recommends that you use only valid SSL certificates that have been issued by a trusted certificate authority.

At the command prompt, type the following commands to bind an SSL certificate-key pair to a virtual server and verify the configuration:

- bind ssl vserver <vServerName> -certkeyName <certificate-KeyPairName> -CA -skipCAName
- show ssl vserver <vServerName>

Example

```
> bind ssl vs vs1 -certkeyName cert2 -CA -skipCAName
Done
> sh ssl vs vs1
Advanced SSL configuration for VServer vs1:
DH: DISABLED
Ephemeral RSA: ENABLED Refresh Count: 0
Session Reuse: ENABLED Timeout: 120 seconds
Cipher Redirect: DISABLED
SSLv2 Redirect: DISABLED
ClearText Port: 0
Client Auth: DISABLED
SSL Redirect: DISABLED
Non FIPS Ciphers: DISABLED
SNI: DISABLED
SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED
```

```
 Push Encryption Trigger: Always
 Send Close-Notify: YES
 1) CertKey Name: cert1 CA Certificate OCSPCheck: Optional CA_Name Sent
 2) CertKey Name: cert2 CA Certificate OCSPCheck: Optional CA_Name Skipped
 1) Cipher Name: DEFAULT
 Description: Predefined Cipher Alias
Done
```

If you try to unbind a certificate-key pair from a virtual server by using the **unbind ssl certKey <certkeyName>** command, an error message appears because the syntax of the command has changed. At the command prompt, type the following command:

unbind ssl vserver <vServerName> -certkeyName <string>

Example

```
unbind ssl vserver vssl -certkeyName sslckey
```

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server to bind the certificate key to, and then click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, click SSL Settings.
4. In the Available pane, select a certificate.
5. Click Add to add the certificate as a server certificate. To add as an SNI certificate, in the Add drop-down list select As SNI. To add as a CA certificate, in the Add drop-down list select As CA.
6. In the Configured pane, to indicate that the CA certificate's CA Name should not be sent to the SSL client, select Skip CA.
7. Click OK. The certificate pair is bound to the virtual server.

Configuring an SSL Virtual Server for Secure Hosting of Multiple Sites

May 13, 2015

Virtual hosting is used by Web servers to host more than one domain name with the same IP address. The NetScaler supports hosting of multiple secure domains by offloading SSL processing from the Web servers using transparent SSL services or virtual server-based SSL offloading. However, when multiple Web sites are hosted on the same virtual server, the SSL handshake is completed before the expected host name is sent to the virtual server. As a result, the NetScaler cannot determine which certificate to present to the client after a connection is established. This problem is resolved by enabling Server Name Indication (SNI) on the virtual server. SNI is a Transport Layer Security (TLS) extension used by the client to provide the host name during handshake initiation. Based on the information provided by the client in the SNI extension, the NetScaler presents the corresponding certificate to the client.

A wildcard SSL Certificate helps enable SSL encryption on multiple subdomains if the domains are controlled by the same organization and share the same second-level domain name. For example, a wildcard certificate issued to a sports network using the common name "*.sports.net" can be used to secure domains, such as "login.sports.net" and "help.sports.net" but not "login.ftp.sports.net."

Note: On a NetScaler appliance, SNI is not supported with a Subject Alternative Name (SAN) extension certificate.
You can bind multiple server certificates to a single SSL virtual server or transparent service using the -SNICert option. These certificates are issued by the virtual server or service if SNI is enabled on the virtual server or service. You can enable SNI at any time.

At the command prompt, type the following commands to configure SNI and verify the configuration:

- set ssl vserver <vServerName>@ [-SNIEnable (ENABLED | DISABLED)]
- bind ssl vserver <vServerName>@ -certkeyName <string> -SNICert
- show ssl vserver <vServerName>

  To bind multiple server certificates to a transparent service by using the NetScaler command line, replace vserver with service and vservername with servicename in the above commands.

  Note: The SSL service should be created with -clearTextPort 80 option.

Example

```
set ssl vserver v1 -snI ENABLED
bind ssl vserver v1 -certkeyName serverabc -SNICert
sh ssl vserver v1
Advanced SSL configuration for VServer v1:
...
SSL Redirect: DISABLED
Non FIPS Ciphers: DISABLED
SNI: ENABLED
```

SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED
1)CertKey Name: servercert Server Certificate
1)CertKey Name: abccert Server Certificate for SNI
2)CertKey Name: xyzcert Server Certificate for SNI
3)CertKey Name: startcert Server Certificate for SNI
1)Cipher Name: DEFAULT
Description: Predefined Cipher Alias
Done

1. Navigate to Traffic Management > SSL Offload > Virtual Servers or Traffic Management > SSL Offload > Services.
2. In the details pane, select the virtual server or service on which SNI is to be enabled, and then click Open.
3. In the Configure Virtual Server (SSL Offload) or Configure Service dialog box, on the SSL Settings tab, click SSL Parameters.
4. In the Configure SSL Params dialog box, under Others, select the SNI Enable check box.
5. Click OK.
6. On the SSL Settings tab, under Available, select a certificate.
7. In the Add drop-down list select As SNI.
8. To add more certificates, repeat step 7.
9. Under Configured, verify that the certificate is added as a server certificate for SNI.
10. Click OK.

Enabling Stricter Control on Client Certificate Validation

Jun 19, 2015

The NetScaler appliance accepts valid Intermediate-CA certificates if they are issued by a single Root-CA. That is, if only the Root-CA certificate is bound to the virtual server, and any intermediate certificate sent with the client certificate is validated by that Root-CA, the appliance trusts the certificate chain and the handshake is successful.

However, if a client sends a chain of certificates in the handshake, none of the intermediate certificates can be validated by using a CRL or OCSP responder unless that certificate is bound to the SSL virtual server. Therefore, even if one of the intermediate certificates is revoked, the handshake is successful. As part of the handshake, the SSL virtual server sends the list of CA certificates that are bound to it. For stricter control, you can configure the SSL virtual server to accept only a certificate that is signed by one of the CA certificates bound to that virtual server. To do so, you must enable the ClientAuthUseBoundCAChain setting in the SSL profile bound to the virtual server. The handshake fails if the client certificate is not signed by one of the CA certificates bound to the virtual server.

For example, say two client certificates, clientcert1 and clientcert2, are signed by the intermediate certificates Int-CA-A and Int-CA-B, respectively. The intermediate certificates are signed by the root certificate Root-CA. Int-CA-A and Root-CA are bound to the SSL virtual server. In the default case (ClientAuthUseBoundCAChain disabled), both clientcert1 and clientcert2 are accepted. However, if ClientAuthUseBoundCAChain is enabled, only clientcert1 is accepted by the NetScaler appliance

To enable stricter control on client certificate validation by using the command line

At the NetScaler command prompt, type:

- set ssl profile <name> -ClientAuthUseBoundCAChain Enabled
- set ssl vserver <vServerName> -sslProfile <string>

To enable stricter control on client certificate validation by using the configuration utility

1. Navigate to System > Profiles, select the SSL Profiles tab, and create an SSL profile, or select an existing profile.
2. Select Enable Client Authentication using bound CA Chain.
3. Navigate to Traffic Management > Load Balancing > Virtual Servers, and select an SSL virtual server.
4. In Advanced Settings, select SSL Profiles, and select the profile on which you enabled Enable Client Authentication using bound CA Chain.
5. Click OK, and then click Done.

Managing Certificates

Jun 03, 2015

An SSL certificate, which is an integral part of any SSL transaction, is a digital data form (X509) that identifies a company (domain) or an individual. The certificate has a public key component that is visible to any client that wants to initiate a secure transaction with the server. The corresponding private key, which resides securely on the NetScaler appliance, is used to complete asymmetric key (or public key) encryption and decryption.

You can obtain an SSL certificate and key in either of the following ways:

- From an authorized certificate authority (CA), such as VeriSign
- By generating a new SSL certificate and key on the NetScaler appliance

Alternately, you can use an existing SSL certificate on the appliance.

Caution: Citrix recommends that you use certificates obtained from authorized CAs, such as VeriSign, for all your SSL transactions. Certificates generated on the NetScaler appliance should be used for testing purposes only, not in any live deployment.

To manage certificates, see the following sections:

- Obtaining a Certificate from a Certificate Authority
- Importing Existing Certificates and Keys
- Generating a Self-Signed Certificate
- Adding a Group of SSL Certificates
- Displaying a Certificate Chain
- Generating a Server Test Certificate
- Modifying and Monitoring Certificates and Keys
- Using Global Site Certificates
- Converting the Format of SSL Certificates for Import or Export

Obtaining a Certificate from a Certificate Authority

Mar 30, 2015

A certificate authority (CA) is an entity that issues digital certificates for use in public key cryptography. Certificates issued or signed by a CA are automatically trusted by applications, such as web browsers, that conduct SSL transactions. These applications maintain a list of the CAs that they trust. If the certificate being used for the secure transaction is signed by any of the trusted CAs, the application proceeds with the transaction.

To obtain an SSL certificate from an authorized CA, you must create a private key, use that key to create a certificate signing request (CSR), and submit the CSR to the CA. The only special characters allowed in the file names are underscore and dot.

The private key is the most important part of a digital certificate. By definition, this key is not to be shared with anyone and should be kept securely on the NetScaler appliance. Any data encrypted with the public key can be decrypted only by using the private key.

The appliance supports two encryption algorithms, RSA and DSA, for creating private keys. You can submit either type of private key to the CA. The certificate that you receive from the CA is valid only with the private key that was used to create the CSR, and the key is required for adding the certificate to the NetScaler.

Caution: Be sure to limit access to your private key. Anyone who has access to your private key can decrypt your SSL data.
All SSL certificates and keys are stored in the /nsconfig/ssl folder on the appliance. For added security, you can use the Data Encryption Standard (DES) or triple DES (3DES) algorithm to encrypt the private key stored on the appliance.
Note: The length of the SSL key name allowed includes the length of the absolute path name if the path is included in the key name.

To create an RSA private key by using the command line interface

At the command prompt, type the following command:

create ssl rsakey <keyFile> <bits> [-exponent (3 | F4)] [-keyform (DER | PEM)]

Example

```
> create ssl rsakey Key-RSA-1 2048 -exponent F4 -keyform PEM
```

To create a DSA private key by using the command line interface

At the command prompt, type the following command:

create ssl dsakey <keyfile> <bits> [-keyform (DER | PEM)]

Example

```
> create ssl dsakey Key-DSA-1 2048 -keyform PEM
```

To create an RSA private key by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. In the details pane, under SSL Keys, click Create RSA Key.
3. In the Create RSA Key dialog box, configure the RSA key parameters. For a description of a parameter, hover the mouse cursor over the corresponding field.
4. Click Create, and then click Close.

To create an DSA private key by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. In the details pane, under SSL Keys, click Create DSA Key.
3. In the Create DSA Key dialog box, configure the DSA key parameters. For a description of a parameter, hover the mouse cursor over the corresponding field.
4. Click Create, and then click Close.

The certificate signing request (CSR) is a collection of information, including the domain name, other important company details, and the private key to be used to create the certificate. To avoid generating an invalid certificate, make sure that the details you provide are accurate.

To create a certificate signing request by using the command line interface

At the command prompt, type the following command:

create ssl certreq <reqFile> -keyFile <input_filename> | -fipsKeyName <string>) [-keyForm (DER | PEM) {-PEMPassPhrase }] -countryName <string> -stateName <string> -organizationName <string> [-organizationUnitName <string>] [-localityName <string>] [-commonName <string>] [-emailAddress <string>] {-challengePassword } [-companyName <string>]

Example

```
create ssl certreq csreq1 -keyfile ramp -keyform PEM -countryName US -stateName Florida -localityName FortLauderdale  -organizationName Citrix -organizationUnitName I
```

To create a certificate signing request by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. In the details pane, under SSL Certificates, click Create CSR (Certificate Signing Request).
3. In the Create CSR (Certificate Signing Request) dialog box, configure the CSR parameters. For a description of a parameter, hover the mouse cursor over the corresponding field.
4. Click Create, and then click Close. The certificate signing request you created is saved on the appliance in the specified location.

Most CAs accept certificate submissions by email. The CA will return a valid certificate to the email address from which you submit the CSR.

Importing Existing Certificates and Keys

Nov 11, 2013

If you want to use certificates and keys that you already have on other secure servers or applications in your network, you can export them, and then import them to the NetScaler appliance. You might have to convert exported certificates and keys before you can import them to the NetScaler appliance.

For the details of how to export certificates from secure servers or applications in your network, see the documentation of the server or application from which you want to export.

Note: For installation on the NetScaler appliance, key and certificate names cannot contain spaces or special characters other than those supported by the UNIX file system. Follow the appropriate naming convention when you save the exported key and certificate.
A certificate and private key pair is commonly sent in the PKCS#12 format. The NetScaler supports PEM and DER formats for certificates and keys. To convert PKCS#12 to PEM or DER, or PEM or DER to PKCS#12, see Converting the Format of SSL Certificates for Import or Export.

The NetScaler appliance does not support PEM keys in PKCS#8 format. However, you can convert these keys to a supported format by using the OpenSSL interface, which you can access from the NetScaler command line or the configuration utility. Before you convert the key, you need to verify that the private key is in PKCS#8 format. Keys in PKCS#8 format typically start with the following text:

-----BEGIN ENCRYPTED PRIVATE KEY-----

leuSSZQZKgrgUQ==

-----END ENCRYPTED PRIVATE KEY-----

1. Open an SSH connection to the appliance by using an SSH client, such as PuTTY.
2. Log on to the appliance by using the administrator credentials.
3. At the command prompt, type shell.
4. At the shell prompt type openssl.

1. Navigate to Traffic Management > SSL.
2. In the details pane, under Tools, click OpenSSL interface.

At the OpenSSL prompt, type one of the following commands, depending on whether the non-supported key format is of type rsa or dsa:

- rsa- in <PKCS#8 Key Filename> -des3 -out <encrypted Key Filename>
- dsa -in <PKCS#8 Key Filename> -des3 -out <encrypted Key Filename>

At the OpenSSL prompt, type the following commands, depending on whether the non-supported key format is of type rsa or dsa:

- rsa -in <PKCS#8 Key Filename> -out <unencrypted Key Filename>
- dsa -in <PKCS#8 Key Filename> -out <unencrypted Key Filename>

Parameters for converting an unsupported key format to a supported key format

<PKCS#8 Key Filename>

The input file name of the incompatible PKCS#8 private key.

<encrypted Key Filename>

The output file name of the compatible encrypted private key in PEM format.

<unencrypted Key Filename>

The output file name of the compatible unencrypted private key in PEM format.

Generating a Self-Signed Certificate

Jun 15, 2015

The NetScaler appliance has a built in CA tools suite that you can use to create self-signed certificates for testing purposes.

Caution: Because these certificates are signed by the NetScaler itself, not by an actual CA, you should not use them in a production environment. If you attempt to use a self-signed certificate in a production environment, users will receive a "certificate invalid" warning each time the virtual server is accessed.
The NetScaler supports creation of the following types of certificates

- Root-CA certificates
- Intermediate-CA certificates
- End-user certificates
  - server certificates
  - client certificates

Before generating a certificate, create a private key and use that to create a certificate signing request (CSR) on the appliance. Then, instead of sending the CSR out to a CA, use the NetScaler CA Tools to generate a certificate.

For details on how to create a private key and a CSR, see Obtaining a Certificate from a Certificate Authority.

1. Navigate to Traffic Management > SSL.
2. In the details pane, under Getting Started, select the wizard for the type of certificate that you want to create.
3. Follow the instructions on the screen.

At the command prompt, type the following command:

create ssl cert <certFile> <reqFile> <certType> [-keyFile <input_filename>] [-keyform (DER | PEM)] [-days <positive_integer>]
Example

```
> create ssl cert certi1 csreq1 ROOT_CERT -keyFile
rsa1 -keyForm PEM -days 365
  Done
```

At the command prompt, type the following command:

create ssl cert <certFile> <reqFile> <certType> [-keyFile <input_filename>] [-keyform (DER | PEM)] [-days <positive_integer>] [-certForm (DER | PEM)] [-CAcert <input_filename>] [-CAcertForm (DER | PEM)] [-CAkey <input_filename>] [-CAkeyForm (DER | PEM)] [-CAserial <output_filename>]
Example

```
> create ssl cert certsy csr1 INTM_CERT -CAcert cert1
-CAkey rsakey1 -CAserial 23
Done
```

1. Navigate to Traffic Management > SSL.
2. Under SSL Certificates, click Create Certificate.
3. In the Create Certificate dialog box, specify values for the following parameters:
   - Certificate File Name*
   - Certificate Format
   - Certificate Type
   - Certificate Request File Name*
   - Key File Name*
   - Key Format
   - PEM Passphrase (For Encrypted Key)—If the key is encrypted, you are prompted to enter the password at run-time on the CLI.
   - Validity Period (Number of Days)

   * A required parameter

   Note: Instead of typing the file name, you can use the browse button to launch the NetScaler file browser and select the file.
4. Click Create, and then click Close. The Root-CA certificate you created is saved on the NetScaler.

1. Navigate to Traffic Management > SSL.
2. Under SSL Certificates, click Create Certificate.
3. In the Create Certificate dialog box, specify values for the following parameters:
   - Certificate File Name*
   - Certificate Format
   - Certificate Type
   - Certificate Request File Name*
   - PEM Passphrase (For Encrypted Key)—If the key is encrypted, you are prompted to enter the password at run-time on the CLI.
   - Validity Period (Number of Days)
   - CA Certificate File Name*
   - CA Certificate File Format
   - CA Key File Name*—CAkey
   - CA Key File Format
   - PEM Passphrase (For Encrypted CA Key)
   - CA Serial Number File*

   * A required parameter

   Note: Instead of typing the file name, you can use the browse button to launch the NetScaler file browser and select the file.
4. Click Create, and then click Close. The Intermediate-CA certificate you created is saved on the NetScaler.

The Diffie-Hellman (DH) key exchange is a way for two parties involved in an SSL transaction that have no prior knowledge of each other to agree upon a shared secret over an insecure channel. This secret can then be converted into cryptographic keying material for mainly symmetric key cipher algorithms that require such a key exchange.

This feature is disabled by default and should be specifically configured to support ciphers that use DH as the key exchange algorithm.
Note: Generating a 2048-bit DH key may take a long time (up to 30 minutes).

To generate a DH key by using the command line interface

At the command prompt, type the following command:

create ssl dhparam <dhFile> [<bits>] [-gen (2 | 5)]

Example

```
create ssl dhparam Key-DH-1 512 -gen 2
```

To generate a DH key by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. Under Tools, click Create Diffe-Hellman (DH) Key.
3. In the Create DH Param dialog box, specify values for the following parameters:
   - DH File Name (with path)*
   - DH Parameter Size (Bits)*
   - DH Generator

   * A required parameter
4. Click OK.

Adding a Group of SSL Certificates

Feb 17, 2017

If the server certificate is issued by an intermediate CA that is not recognized by standard web browsers as a trusted CA, the CA certificate(s) must be sent to the client with the server's own certificate. Otherwise, the browser terminates the SSL session because it fails to authenticate the server certificate.

There are two ways to add the server and intermediate certificates:

- Create a certificate set that contains the chain of certificates.
- Create a chain of certificates manually by adding and linking the certificates individually.

Note: This feature is not supported on the NetScaler FIPS platform and in a cluster setup.
Instead of adding and linking individual certificates, you can now group a server certificate and up to nine intermediate certificates in a single file, and then specify the file's name when adding a certificate-key pair. Before you do so, make sure that the following prerequisites are met.

- The certificates in the file are in the following order:
  - Server certificate (should be the first certificate in the file)
  - Optionally, a server key
  - Intermediate certificate 1 (ic1)
  - Intermediate certificate 2 (ic2)
  - Intermediate certificate 3 (ic3), and so on
    Note: Intermediate certificate files are created for each intermediate certificate with the name "<certificatebundlename>.pem_ic<*n*>" where n is between 1 and 9. For example, bundle.pem_ic1, where bundle is the name of the certificate set and ic1 is the first intermediate certificate in the set.
- Bundle option is selected.
- No more than nine intermediate certificates are present in the file.

The file is parsed and the server certificate, intermediate certificates, and server key (if present) are identified. First, the server certificate and key are added. Then, the intermediate certificates are added, in the order in which they were added to the file, and linked accordingly.

An error is reported if any of the following conditions exist:

- A certificate file for one of the intermediate certificates already exists on the appliance.
- The key is placed before the server certificate in the file.
- An intermediate certificate is placed before the server certificate.
- Intermediate certificates are not in placed in the file in the same order as they are created.
- No certificates are present in the file.
- A certificate is not in the proper PEM format.
- The number of intermediate certificates in the file exceeds nine.

To add a certificate set by using the command line interface

At the command prompt, type the following commands to create a certificate set and verify the configuration:

1. add ssl certKey <certkeyName> -cert <string> -key <string> -bundle (YES | NO)
2. show ssl certKey
3. show ssl certlink

In the following example, the certificate set (bundle.pem) contains the following files:

- server certificate (bundle) linked to bundle_ic1
- First intermediate certificate (bundle_ic1) linked to bundle_ic2
- Second intermediate certificate (bundle_ic2) linked to bundle_ic3
- Third intermediate certificate (bundle_ic3)

```
> add ssl certKey bundle -cert bundle.pem -key bundle.pem -bundle yes

> show ssl certkey

1) Name: bundle
Cert Path: /nsconfig/ssl/bundle.pem
Format: PEM
Status: Valid, Days to expiration:10415
Certificate Expiry Monitor: DISABLED

2) Name: bundle_ic1
Cert Path: /nsconfig/ssl/bundle.pem_ic1
Format: PEM
Status: Valid, Days to expiration:10415
Certificate Expiry Monitor: DISABLED

3) Name: bundle_ic2
Cert Path: /nsconfig/ssl/bundle.pem_ic2
Format: PEM
Status: Valid, Days to expiration:10415
Certificate Expiry Monitor: DISABLED

4) Name: bundle_ic3
Cert Path: /nsconfig/ssl/bundle.pem_ic3
Format: PEM
Status: Valid, Days to expiration:10415
Certificate Expiry Monitor: DISABLED
Done

> show ssl certlink

1) Cert Name: bundle CA Cert Name: bundle_ic1
2) Cert Name: bundle_ic1 CA Cert Name: bundle_ic2
3) Cert Name: bundle_ic2 CA Cert Name: bundle_ic3
Done
```

To add a certificate set by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. In the SSL Certificates pane, click Install.
3. In the Install Certificate dialog box, type the details, such as the certificate and key file name, and then select Certificate Bundle.
4. Click Install, and then click Close.

Updated: 2013-08-20

Instead of using a set of certificates (a single file), you can create a chain of certificates. The chain links the server certificate to its issuer (the intermediate CA). For this approach to work, the intermediate CA certificate file must already be installed on the NetScaler appliance, and one of the certificates in the chain must be trusted by the client application. For example, link Cert-Intermediate-A to Cert-Intermediate-B, where Cert-Intermediate-B is linked to Cert-Intermediate-C, which is a certificate trusted by the client application.

Note: The NetScaler supports sending a maximum of 10 certificates in the chain of certificates sent to the client (one server certificate and nine CA certificates).

To create a certificate chain by using the command line interface

At the command prompt, type the following commands to create a certificate chain and verify the configuration. (Repeat the first command for each new link in the chain.)

- link ssl certkey <certKeyName> <linkCertKeyName>
- show ssl certlink

Example

```
> link ssl certkey siteAcertkey CAcertkey
 Done

> show ssl certlink

linked certificate:
      1) Cert Name: siteAcertkey CA Cert Name: CAcertkey
 Done
```

To create a certificate chain by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. Select the server certificate you want to link, and then click Link.
3. In Link Server Certificate(s), select CA Certificate Name to be linked to.
4. Click OK. The server certificate is now linked to the intermediate certificate.

Generating a Server Test Certificate

Aug 20, 2013

The NetScaler appliance allows you to create a test certificate for server authentication by using a GUI wizard in the configuration utility. A server certificate is used to authenticate and identify a server in an SSL handshake. A server certificate is generally issued by a trusted CA and is sent out by the server to a client who uses it to authenticate the server.

For issuing a server test certificate, the appliance operates as a CA. This certificate can be bound to an SSL virtual server for authentication in an SSL handshake with a client. This certificate is for testing purposes only. It should not be used in a production environment.

You can install the server test certificate on any virtual server that uses the SSL or the SSL_TCP protocol.

1. Navigate to Traffic Management > SSL.
2. Under SSL Certificates, click Create and install a Server Test Certificate.
3. In the Create and install a Server Test Certificate dialog box, specify values for the following parameters:
   - Certificate File Name—name of the server test certificate
   - Fully Qualified Domain Name—the domain for which you want to secure the connection
   - Country—the name of the country or region
4. Click OK.

Modifying and Monitoring Certificates and Keys

Apr 28, 2016

To avoid downtime when replacing a certificate-key pair, you can update an existing certificate. If you want to replace a certificate with a certificate that was issued to a different domain, you must disable domain checks before updating the certificate.

To receive notifications about certificates due to expire, you can enable the expiry monitor.

Updating an Existing Server Certificate

When you remove or unbind a certificate from a configured SSL virtual server, or an SSL service, the virtual server or service becomes inactive until a new valid certificate is bound to it. To avoid downtime, you can use the update feature to replace a certificate-key pair that is bound to an SSL virtual server or an SSL service, without first unbinding the existing certificate.

To update an existing certificate-key pair by using the command line interface

At the command prompt, type the following commands to update an existing certificate-key pair and verify the configuration:

- update ssl certkey <certkeyName> -cert <string> -key <string>
- show ssl certKey <certkeyName>

Example

```
> update ssl certkey siteAcertkey -cert /nsconfig/ssl/cert.pem
     -key /nsconfig/ssl/pkey.pem
  Done

> show ssl certkey siteAcertkey
Name: siteAcertkey      Status: Valid
     Version: 3
     Serial Number: 02
     Signature Algorithm: md5WithRSAEncryption
     Issuer: /C=US/ST=CA/L=Santa Clara/O=siteA/OU=Tech
     Validity
        Not Before: Nov 11 14:58:18 2001 GMT
        Not After: Aug 7 14:58:18 2004 GMT
     Subject: /C=US/ST-CA/L=San Jose/O=CA/OU=Security
     Public Key Algorithm: rsaEncryption
     Public Key size: 2048
  Done
```

To update an existing certificate-key pair by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. Select the certificate you want to update, and then click Update.
3. Use the Browse button next to the Certificate File name and the Key File name and select the new certificate and key

files respectively.

4. If the key in encrypted, in the Password text box, type the password used to encrypt the key.
5. Click OK. In SSL Certificates pane, select the certificate that you just updated and verify that the settings displayed at the bottom of the screen are correct.

Disabling Domain Checks

When an SSL certificate is replaced on the NetScaler, the domain name mentioned on the new certificate should match the domain name of the certificate being replaced. For example, if you have a certificate issued to abc.com, and you are updating it with a certificate issued to def.com, the certificate update fails.

However, if you want the server that has been hosting a particular domain to now host a new domain, you can disable the domain check before updating its certificate.

To disable the domain check for a certificate by using the command line interface

At the command prompt, type the following commands to disable the domain check and verify the configuration:

- update ssl certKey <certkeyName> -noDomainCheck
- show ssl certKey <certkeyName>

Example

```
> update ssl certKey sv -noDomainCheck
 Done
> show ssl certkey sv
 Name: sv
 Cert Path: /nsconfig/ssl/complete/server/server_rsa_512.pem
 Key Path: /nsconfig/ssl/complete/server/server_rsa_512.ky
 Format: PEM
 Status: Valid,  Days to expiration:9349
 Certificate Expiry Monitor: DISABLED
 Done
```

To disable the domain check for a certificate by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. Select the certificate you want to update, and then click Update.
3. Select No Domain Check, and then click OK. The domain check for the certificate is now disabled.

Enabling the Expiry Monitor

An SSL certificate is valid for a specific period of time. A typical deployment includes multiple virtual servers that process SSL transactions, and the certificates bound to them can expire at different times. An expiry monitor configured on the NetScaler appliance creates entries in the appliance's syslog and nsaudit logs when a certificate configured on the appliance is due to expire.

If you want to create SNMP alerts for certificate expiration, you must configure them separately.

To enable an expiry monitor for a certificate by using the command line interface

At the command prompt, type the following commands to enable an expiry monitor for a certificate and verify the configuration:

- set ssl certKey <certkeyName> [-expiryMonitor (ENABLED | DISABLED) [-notificationPeriod <positive_integer>]]
- show ssl certKey <certkeyName>

Example

```
> set ssl certKey sv -expiryMonitor ENABLED –notificationPeriod 60
 Done

> show ssl certkey sv
Name: sv
 Cert Path: /nsconfig/ssl/complete/server/server_rsa_512.pem
 Key Path: /nsconfig/ssl/complete/server/server_rsa_512.ky
 Format: PEM
 Status: Valid,  Days to expiration:9349
 Certificate Expiry Monitor: ENABLED
 Expiry Notification period: 60 days
 Done
```

To enable an expiry monitor for a certificate by using the configuration utility

1. Navigate to Traffic Management > SSL > Certificates.
2. Select the certificate you want to update, and then click Update.
3. Select the Enable option.
4. In the Notification Period text box, type the required notification period value.
   Note: The notification period parameter can be set to any value between 10 and 100 days and the default notification period is 30 days.
5. Click OK. In the SSL Certificates pane, select the certificate that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Using Global Site Certificates

Nov 11, 2013

A global site certificate is a special-purpose server certificate whose key length is greater than 128 bits. A global site certificate consists of a server certificate and an accompanying intermediate-CA certificate. You must import the global site certificate and its key from the server to the NetScaler appliance.

How Global Site Certificates Work

Export versions of browsers use 40-bit encryption to initiate connections to SSL Web-servers. The server responds to connection requests by sending its certificate. The client and server then decide on an encryption strength based on the server certificate type:

- If the server certificate is a normal certificate and not a global site certificate, the export client and server complete the SSL handshake and uses 40-bit encryption for data transfer.
- If the server certificate is a global site certificate (and if the export client feature is supported by the browser), the export client automatically upgrades to 128-bit encryption for data transfer.

If the server certificate is a global site certificate, the server sends its certificate, along with the accompanying intermediate-CA certificate. The browser first validates the intermediate-CA certificate by using one of the Root-CA certificates that are normally included in web browsers. Upon successful validation of the intermediate-CA certificate, the browser uses the intermediate-CA certificate to validate the server certificate. Once the server is successfully validated, the browser renegotiates (upgrades) the SSL connection to 128-bit encryption.

With Microsoft's Server Gated Cryptography (SGC), if the Microsoft IIS server is configured with an SGC certificate, export clients that receive the certificate renegotiate to use128-bit encryption.

Importing a Global Site Certificate

To import a global site certificate, first export the certificate and server key from the Web server. Global site certificates are generally exported in some binary format, therefore, before importing the global site certificate, convert the certificate and key to the PEM format.

To import a global site certificate

1. Using a text editor, copy the server certificate and the accompanying intermediate-CA certificate into two separate files.

   The individual PEM encoded certificate will begin with the header ----- BEGIN CERTIFICATE----- and end with the trailer -----END CERTIFICATE-----.

2. Use an SFTP client to transfer the server certificate, intermediate-CA certificate, and server-key to the NetScaler.
3. Use the following OpenSSL command to identify the server certificate and intermediate-CA certificate from the two separate files.
   Note: You can launch the OpenSSL interface from the configuration utility. In the navigation pane, click SSL. In the details pane, under Tools, click Open SSL interface.

   ```
   > openssl x509 -in >path of the CA cert file< -text
   ```

X509v3 Basic Constraints:
CA:TRUE
X509v3 Key Usage:
Certificate Sign, CRL Sign
Netscape Cert Type:
SSL CA, S/MIME CA

> openssl x509 -in >path of the server certificate file< -text

X509v3 Basic Constraints:
CA:FALSE
Netscape Cert Type:
SSL Server

4. At the FreeBSD shell prompt, enter the following command:

   openssl x509 -in cert.pem -text | more

   Where **cert.pem** is one of the two certificate files.

   Read the **Subject** field in the command output. For example,

   Subject: C=US, ST=Oregon, L=Portland,
   O=mycompany, Inc., OU=IT, CN=www.mycompany.com
   If the CN field in the Subject matches the domain-name of your Web site, then this is the server certificate and the other certificate is the accompanying intermediate-CA certificate.

5. Use the server certificate and its private key) to create a certificate key pair on the NetScaler. For details on creating a certificate-key pair on the NetScaler, see Adding a Certificate Key Pair.
6. Add the intermediate-CA certificate on the NetScaler. Use the server certificate you created in step 4 to sign this intermediate certificate. For details on creating an Intermediate-CA certificate on the NetScaler, see Generating a Self-Signed Certificate.

Converting the Format of SSL Certificates for Import or Export

Aug 20, 2013

A NetScaler appliance supports the PEM and DER formats for SSL certificates. Other applications, such as client browsers and some external secure servers, require various public key cryptography standard (PKCS) formats. The NetScaler can convert the PKCS#12 format (the personal information exchange syntax standard) to PEM or DER format for importing a certificate to the appliance, and can convert PEM or DER to PKCS#12 for exporting a certificate. For additional security, conversion of a file for import can include encryption of the private key with the DES or DES3 algorithm.

Note: If you use the configuration utility to import a PKCS#12 certificate, and the password contains a dollar sign ($), backquote (`), or escape (\) character, the import may fail. If it does, the ERROR: Invalid password message appears. If you must use a special character in the password, be sure to prefix it with an escape character (\) unless all imports are performed by using the NetScaler command line.

To convert the format of a certificate by using the command line interface

At the command prompt, type the following command:

Convert ssl pkcs12 <outfile> [-import [-pkcs12File <inputFilename>] [-des | -des3] [-export [-certFile <inputFilename>] [-keyFile <inputFilename>]] During the operation, you are prompted to enter an import password or an export password. For an encrypted file, you are also prompted to enter a passphrase.

Example

```
convert ssl pkcs12 Cert-Import-1.pem -import -pkcs12File Cert-Import-1.pfx -des

convert ssl pkcs12 Cert-Client-1.pfx -export -certFile Cert-Client-1 -keyFile Key-Client-1
```

To convert the format of a certificate by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. Under Tools, do one of the following
   - To convert a PKCS#12 certificate and key to PEM format, click Import PKCS#12.
   - To convert a certificate and key from PEM to PKCS#12 format, click Export PKCS#12.
3. In the Import PKCS12 or Export PKCS12 dialog box, set the following parameters:
   - Output File Name*
   - PKCS12 File Name*
   - Certificate File Name*
   - Key File Name*
   - Import Password*
   - Export Password*
   - Encoding Format
   - PEM Passphrase

   * A required parameter
4. Click OK.

Managing Certificate Revocation Lists

Jun 03, 2015

A certificate issued by a CA typically remains valid until its expiration date. However, in some circumstances, the CA may revoke the issued certificate before the expiration date (for example, when an owner's private key is compromised, a company's or individual's name changes, or the association between the subject and the CA changes).

A Certificate Revocation List (CRL) identifies invalid certificates by serial number and issuer.

Certificate authorities issue CRLs on a regular basis. You can configure the NetScaler appliance to use a CRL to block client requests that present invalid certificates.

If you already have a CRL file from a CA, add that to the NetScaler. You can configure refresh options. You can also configure the NetScaler to sync the CRL file automatically at a specified interval, from either a web location or an LDAP location. The NetScaler supports CRLs in either the PEM or the DER file format. Be sure to specify the file format of the CRL file being added to the NetScaler.

If you have used the NetScaler as a CA to create certificates that are used in SSL deployments, you can also create a CRL to revoke a particular certificate. This feature can be used, for example, to ensure that self-signed certificates that are created on the NetScaler are not used either in a production environment or beyond a particular date.

Note:
By default, CRLs are stored in the /var/netscaler/ssl directory on the NetScaler appliance.

To manage certificate revocation lists, see the following sections:

- Creating a CRL on the NetScaler
- Adding an Existing CRL to the NetScaler
- Configuring CRL Refresh Parameters
- Synchronizing CRLs
- Performing Client Authentication by using a Certificate Revocation List

Creating a CRL on the NetScaler

Updated: 2013-09-04

Since you can use the NetScaler appliance to act as a certificate authority and create self-signed certificates, you can also revoke certificates that you have created and certificates whose CA certificate you own.

The appliance must revoke invalid certificates before creating a CRL for those certificates. The appliance stores the serial numbers of revoked certificates in an index file and updates the file each time it revokes a certificate. The index file is automatically created the first time a certificate is revoked.

To revoke a certificate or create a CRL by using the command line interface

At the command prompt, type the following command:

create ssl crl <CAcertFile> <CAkeyFile> <indexFile> (-revoke <input_filename> | -genCRL <output_filename>)
Example

```
create ssl crl Cert-CA-1 Key-CA-1 File-Index-1 -revoke Invalid-1

create ssl crl Cert-CA-1 Key-CA-1 File-Index-1 -genCRL CRL-1
```

To revoke a certificate or create a CRL by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. Under Getting Started, click CRL Management.
3. In the CRL Management dialog box, set the following parameters:
   - CA Certificate File Name*
   - CA Key File Name*
   - CA Key File Password—the password used to encrypt the key file. On the CLI, you are prompted to enter this password at run time.
   - Index File Name*
   - Choose Operation-
     - Revoke Certificate
     - Generate CRL

   * A required parameter
4. Click Create, and then click Close.

Adding an Existing CRL to the NetScaler

Updated: 2013-09-05

Before you configure the CRL on the NetScaler appliance, make sure that the CRL file is stored locally on the NetScaler. In the case of an HA setup, the CRL file must be present on both NetScaler appliances, and the directory path to the file must be the same on both appliances.

To add a CRL on the NetScaler by using the command line

At the command prompt, type the following commands to add a CRL on the NetScaler and verify the configuration:

- add ssl crl <crlName> <crlPath> [-inform (DER | PEM)]
- show ssl crl [<crlName>]

Example

```
> add ssl crl crl-one /var/netscaler/ssl/CRL-one -inform PEM
```

```
 Done
> show ssl crl crl-one
        Name: crl-one   Status: Valid,  Days to expiration: 29
        CRL Path: /var/netscaler/ssl/CRL-one
        Format: PEM     CAcert: samplecertkey
        Refresh: DISABLED
        Version: 1
        Signature Algorithm: sha1WithRSAEncryption
        Issuer:  C=US,ST=California,L=Santa Clara,O=NetScaler Inc.,OU=SSL Acceleration,CN=www.ns.com/emailAddress=support@netscaler.com
        Last_update:Jun 15 10:53:53 2010 GMT
        Next_update:Jul 15 10:53:53 2010 GMT

1)      Serial Number: 00
        Revocation Date:Jun 15 10:51:16 2010 GMT
 Done
```

To add a CRL on the NetScaler by using the configuration utility

1. Navigate to Traffic Management > SSL > CRL.
2. In the details pane, click Add.
3. In the Add CRL dialog box, specify values for the following parameters:
   - CRL Name*
   - CRL File*
   - Format
   - CA Certificate

   * A required parameter
4. Click Create, and then click Close. In the CRL pane, select the CRL that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Configuring CRL Refresh Parameters

Updated: 2014-09-29

A CRL is generated and published by a Certificate Authority periodically or, in some cases, immediately after a particular certificate is revoked. Citrix recommends that you update CRLs on the NetScaler appliance regularly, for protection against clients trying to connect with certificates that are not valid.

The NetScaler can refresh CRLs from a web location or an LDAP directory. When you specify refresh parameters and a web location or an LDAP server, the CRL does not have to be present on the local hard disk drive at the time you execute the command. The first refresh stores a copy on the local hard disk drive, in the path specified by the CRL File parameter. The default path for storing the CRL is /var/netscaler/ssl.
Note: In release 10.0 and later, the method for refreshing a CRL is not included by default. You must explicitly specify an HTTP or LDAP method. If you are upgrading from an earlier release to release 10.0 or later, you must add a method and run the command again.

To configure CRL autorefresh by using the command line

At the command prompt, type the following commands to configure CRL auto refresh and verify the configuration the following commands to configure CRL auto refresh and verify the configuration:

- set ssl crl <crlName> [-refresh (ENABLED | DISABLED)] [-CAcert <string>] [-url <URL | -server <ip_addr|ipv6_addr>] [-method HTTP | (LDAP [-baseDN <string>] [-bindDN <string>] [-scope (Base | One)] [-password <string>] [-binary (YES | NO)])] [-port <port>] [-interval <interval>]
- show ssl crl [<crlName>]

Example

```
Set CRL crl1  -refresh enabled -method ldap -inform DER -CAcert ca1 -server 10.102.192.192 -port 389 -scope base -baseDN "cn=cInt_rsa4_multicert_der,ou=eng,o=ns,c=ir

set ssl crl crl1 -refresh enabled -method http -cacert ca1 -port 80 -time 00:10 -url http://10.102.192.192/crl/ca1.crl

> sh crl

1)      Name: crl1      Status: Valid,     Days to expiration: 355
        CRL Path: /var/netscaler/ssl/crl1
        Format: PEM     CAcert: ca1
        Refresh: ENABLED        Method: HTTP
        URL: http://10.102.192.192/crl/ca1.crl              Port:80
        Refresh Time: 00:10
        Last Update: Successful, Date:Tue Jul  6 14:38:13 2010
Done
```

To configure CRL autorefresh using LDAP or HTTP by using the configuration utility

1. Navigate to Traffic Management > SSL > CRL.
2. Select the configured CRL for which you want to update refresh parameters, and then click Open.
3. Select the Enable CRL Auto Refresh option.
4. In the CRL Auto Refresh Parameters group, specify values for the following parameters:
   - Method
   - Binary
   - Scope
   - Server IP
   - Port*
   - URL
   - Base DN*

- Bind DN
- Password
- Interval
- Day(s)
- Time

* A required parameter

Note: If the new CRL has been refreshed in the external repository before its actual update time as specified by the LastUpdate field of the CRL, you should immediately refresh the CRL on the NetScaler.

5. Click Create. In the CRL pane, select the CRL that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Synchronizing CRLs

Updated: 2013-09-03

The NetScaler appliance uses the most recently distributed CRL to prevent clients with revoked certificates from accessing secure resources.

If CRLs are updated often, the NetScaler needs an automated mechanism to fetch the latest CRLs from the repository. You can configure the NetScaler to update CRLs automatically at a specified refresh interval.

The NetScaler maintains an internal list of CRLs that need to be updated at regular intervals. At these specified intervals, the appliance scans the list for CRLs that need to be updated, connects to the remote LDAP server or HTTP server, retrieves the latest CRLs, and then updates the local CRL list with the new CRLs.

Note: If CRL check is set to mandatory when the CA certificate is bound to the virtual server, and the initial CRL refresh fails, all client-authentication connections with the same issuer as the CRL are rejected as REVOKED until the CRL is successfully refreshed.
You can specify the interval at which the CRL refresh should be carried out. You can also specify the exact time.

To synchronize CRL autorefresh by using the command line interface

At the command prompt, type the following command:

set ssl crl <crlName> [-interval <interval>] [-day <integer>] [-time <HH:MM>]
Example

```
set ssl crl CRL-1 -refresh ENABLE -interval
MONTHLY -days 10 -time 12:00
```

To synchronize CRL refresh by using the configuration utility

1. Navigate to Traffic Management > SSL > CRL.
2. Select the configured CRL for which you want to update refresh parameters, and then click Open.
3. Select the Enable CRL Auto Refresh option.
4. In the CRL Auto Refresh Parameters group, specify values for the following parameters:
   - Interval
   - Day(s)
   - Time
5. Click Create. In the CRL pane, select the CRL that you just configured and verify that the settings displayed at the bottom of the screen are correct.

Performing Client Authentication by using a Certificate Revocation List

Updated: 2013-09-03

If a certificate revocation list (CRL) is present on a NetScaler appliance, a CRL check is performed regardless of whether performing the CRL check is set to mandatory or optional.
The success or failure of a handshake depends on a combination of the following factors:

- Rule for CRL check
- Rule for client certificate check
- State of the CRL configured for the CA certificate

The following table lists the results of the possible combinations for a handshake involving a revoked certificate.

Table 1. Result of a Handshake with a Client Using a Revoked Certificate

Rule for CRL Check	Rule for Client Certificate Check	State of the CRL Configured for the CA certificate	Result of a Handshake with a Revoked Certificate
Optional	Optional	Missing	Success
Optional	Mandatory	Missing	Success
Optional	Mandatory	Present	Failure
Mandatory	Optional	Missing	Success
Mandatory	Mandatory	Missing	Failure
Mandatory	Optional	Present	Success
Mandatory	Mandatory	Present	Failure

Rule for CRL Check	Rule for Client Certificate Check	State of the CRL Configured for the CA certificate	Result of a Handshake with a Revoked Certificate
Optional/Mandatory	Optional	Expired	Success
Optional/Mandatory	Mandatory	Expired	Failure

Note:

- The CRL check is optional by default. To change from optional to mandatory or vice-versa, you must first unbind the certificate from the SSL virtual server, and then bind it again after changing the option.
- In the output of the sh ssl vserver command, OCSP check: optional implies that a CRL check is also optional. The CRL check settings are displayed in the output of the sh ssl vserver command only if CRL check is set to mandatory. If CRL check is set to optional, the CRL check details do not appear.

To configure CRL check by using the command line interface

At the command prompt, type the following command:

bind ssl vserver <vServerName> -certkeyName <string> [(-CA -crlCheck (Mandatory | Optional))]

Example

```
bind ssl vs v1 -certkeyName ca -CA -crlCheck mandatory
sh ssl vserver
> sh ssl vs v1

Advanced SSL configuration for VServer v1:

DH: DISABLED
Ephemeral RSA: ENABLED Refresh Count: 0
Session Reuse: ENABLED Timeout: 120 seconds
Cipher Redirect: DISABLED
SSLv2 Redirect: DISABLED
ClearText Port: 0
Client Auth: ENABLED Client Cert Required: Mandatory
SSL Redirect: DISABLED
Non FIPS Ciphers: DISABLED
SNI: DISABLED
SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED
Push Encryption Trigger: Always
Send Close-Notify: YES

1) CertKey Name: ca CA Certificate CRLCheck: Mandatory CA_Name Sent

1) Cipher Name: DEFAULT
Description: Predefined Cipher Alias
Done
```

To configure CRL check by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, select a virtual server, and then click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, click SSL Settings.
4. In the Configured pane, in the **Check** drop-down list, select CRL Mandatory.
5. Click OK.

Monitoring Certificate Status with OCSP

Jun 03, 2015

Online Certificate Status Protocol (OCSP) is an Internet protocol that is used to determine the status of a client SSL certificate. NetScaler appliances support OCSP as defined in RFC 2560. OCSP offers significant advantages over certificate revocation lists (CRLs) in terms of timely information. Up-to-date revocation status of a client certificate is especially useful in transactions involving large sums of money and high-value stock trades. It also uses fewer system and network resources. NetScaler implementation of OCSP includes request batching and response caching.

To monitor certificate status with OCSP, see the following sections:

- NetScaler Implementation of OCSP
- OCSP Request Batching
- OCSP Response Caching
- Configuring an OCSP Responder

NetScaler Implementation of OCSP

OCSP validation on a NetScaler appliance begins when the appliance receives a client certificate during an SSL handshake. To validate the certificate, the NetScaler creates an OCSP request and forwards it to the OCSP responder. To do so, the NetScaler uses a locally configured URL. The transaction is in a suspended state until the NetScaler evaluates the response from the server and determines whether to allow the transaction or reject it. If the response from the server is delayed beyond the configured time and no other responders are configured, the NetScaler will allow the transaction or display an error, depending on whether the OCSP check was set to optional or mandatory, respectively.

The NetScaler supports batching of OCSP requests and caching of OCSP responses to reduce the load on the OCSP responder and provide faster responses.

OCSP Request Batching

Each time the NetScaler receives a client certificate, it sends a request to the OCSP responder. To help avoid overloading the OCSP responder, the NetScaler can query the status of more than one client certificate in the same request. For this to work efficiently, a timeout needs to be defined so that processing of a single certificate is not inordinately delayed while waiting to form a batch.

OCSP Response Caching

Caching of responses received from the OCSP responder enables faster responses to the clients and reduces the load on the OCSP responder. Upon receiving the revocation status of a client certificate from the OCSP responder, the NetScaler caches the response locally for a predefined length of time. When a client certificate is received during an SSL handshake, the NetScaler first checks its local cache for an entry for this certificate. If an entry is found that is still valid (within the cache timeout limit), it is evaluated and the client certificate is accepted or rejected. If a certificate is not found, the NetScaler sends a request to the OCSP responder and stores the response in its local cache for a configured length of time.

Configuring an OCSP Responder

Updated: 2013-09-05

Configuring OCSP involves adding an OCSP responder, binding the OCSP responder to a certification authority (CA) certificate, and binding the certificate to an SSL virtual server. If you need to bind a different certificate to an OCSP responder that has already been configured, you need to first unbind the responder and then bind the responder to a different certificate.

To add an OCSP responder by using the command line interface

At the command prompt, type the following commands to configure OCSP and verify the configuration:

- add ssl ocspResponder <name> -url <URL> [-cache (ENABLED | DISABLED)[-cacheTimeout <positive_integer>]] [-batchingDepth <positive_integer>][-batchingDelay <positive_integer>] [-resptimeout <positive_integer>] [-responderCert <string> | -trustResponder] [-producedAtTimeSkew <positive_integer>][-signingCert <string>][-useNonce (YES | NO)][-insertClientCert(YES | NO)]
- bind ssl certKey [<certkeyName>] [-ocspResponder <string>] [-priority <positive_integer>]
- bind ssl vserver <vServerName>@ (-certkeyName <string> (CA [-ocspCheck (Mandatory | Optional)]))
- show ssl ocspResponder [<name>]

Example

```
add ssl ocspResponder ocsp_responder1 -url "http:// www.myCA.org:80/ocsp/" -cache ENABLED -cacheTimeout 30 -batchingDepth 8 -batchingDelay 100 -resptimeout 100 -re
bind ssl certKey ca_cert -ocspResponder ocsp_responder1 -priority 1
bind ssl vserver vs1 -certkeyName ca_cert -CA -ocspCheck Mandatory

sh ocspResponder ocsp_responder1
1)Name: ocsp_responder1
URL: http://www.myCA.org:80/ocsp/, IP: 192.128.22.22
Caching: Enabled        Timeout: 30 minutes
Batching: 8 Timeout: 100 mS
HTTP Request Timeout: 100mS
Request Signing Certificate: sign_cert
Response Verification: Full, Certificate: responder_cert
ProducedAt Time Skew: 300 s
Nonce Extension: Enabled
 Client Cert Insertion: Enabled
Done

show certkey ca_cert
Name: ca_cert     Status: Valid,   Days to expiration:8907
Version: 3
...
1)  VServer name: vs1     CA Certificate
1)  OCSP Responder name: ocsp_responder1     Priority: 1
Done
```

```
sh ssl vs vs1
Advanced SSL configuration for VServer vs1:
DH: DISABLED
...
1) CertKey Name: ca_cert CA Certificate OCSPCheck: Mandatory
1) Cipher Name: DEFAULT
  Description: Predefined Cipher Alias
Done
```

To modify an OCSP responder by using the command line interface

You cannot modify the responder name. All other parameters can be changed using the set ssl ocspResponder command.

At the command prompt, type the following commands to set the parameters and verify the configuration:

- set ssl ocspResponder <name> [-url <URL>] [-cache (ENABLED | DISABLED)] [-cacheTimeout <positive_integer>] [-batchingDepth <positive_integer>] [-batchingDelay <positive_integer>] [-resptimeout <positive_integer>] [-responderCert <string> | -trustResponder][-producedAtTimeSkew <positive_integer>][-signingCert <string>] [-useNonce (YES | NO)]
- unbind ssl certKey [<certkeyName>] [-ocspResponder <string>]
- bind ssl certKey [<certkeyName>] [-ocspResponder <string>] [-priority <positive_integer>]
- show ssl ocspResponder [<name>]

To configure an OCSP responder by using the configuration utility

1. Navigate to Traffice Management > SSL > OCSP Responder
2. In the details pane, do one of the following:
   - To create a new responder, click Add.
   - To modify an existing responder, select the responder, and then click Open.
3. Click Create or OK, and then click Close.
4. In the OCSP Responder pane, click the responder that you just configured and verify that the settings displayed at the bottom of the screen are correct.
5. In the navigation pane, click Certificates.
6. In the details pane, select a certificate and click OCSP Bindings.
7. To bind a different certificate-key pair, click Unbind OCSP Responder, and then click Insert OCSP Responder and select a name from the OCSP Responder Name drop-down list. Verify that the settings displayed at the bottom of the screen are correct.
8. Click OK.
9. Navigate to Traffic Management > SSL Offload > Virtual Servers.
10. Select the virtual server to bind the certificate key to, and click Open.
11. In the Configure Virtual Server (SSL Offload) dialog box, click SSL Settings.
12. In the Available pane, select a certificate.
13. In the Add drop-down list select As CA.
14. To make OCSP check mandatory, in the Configured pane, in the Check drop-down list, select OCSP Mandatory.
15. Click OK.

Configuring Client Authentication

Jun 03, 2015

In a typical SSL transaction, the client that is connecting to a server over a secure connection checks the validity of the server by checking the server's certificate before initiating the SSL transaction. In some cases, however, you might want to configure the server to authenticate the client that is connecting to it.

With client authentication enabled on an SSL virtual server, the NetScaler appliance asks for the client certificate during the SSL handshake. The appliance checks the certificate presented by the client for normal constraints, such as the issuer signature and expiration date.

Note: For the NetScaler to verify issuer signatures, the certificate of the CA that issued the client certificate must be installed on the NetScaler and bound to the virtual server that the client is transacting with.
If the certificate is valid, the NetScaler allows the client to access all secure resources. But if the certificate is invalid, the NetScaler drops the client request during the SSL handshake.

The NetScaler verifies the client certificate by first forming a chain of certificates, starting with the client certificate and ending with the root CA certificate for the client (for example, VeriSign). The root CA certificate may contain one or more intermediate CA certificates (if the client certificate is not directly issued by the root CA).

Before you enable client authentication on the NetScaler, make sure that a valid client certificate is installed on the client. Then, enable client authentication for the virtual server that will handle the transactions. Finally, bind the certificate of the CA that issued the client certificate to the virtual server on the NetScaler.

Note: The NetScaler appliance supports a certificate-key pair size of 512 to 4096 bits. The certificate must be signed by using one of the following hash algorithms:

- MD5
- SHA-1
- SHA-224
- SHA-256
- SHA-384
- SHA-512

A NetScaler virtual appliance supports certificates of up to the following sizes:

- 4096-bit server certificate on the virtual server
- 4096-bit client certificate on the service
- 4096-bit CA certificate
- 2048-bit certificate on the physical server
- 2048-bit client certificate (if client authentication is enabled on the virtual server)

To configure client authentication, see the following sections:

- Providing the Client Certificate
- Enabling Client-Certificate-Based Authentication
- Binding CA Certificates to the Virtual Server

Providing the Client Certificate

Before you configure client authentication, a valid client certificate must installed on the client. A client certificate includes details about the specific client system that will create secure sessions with the NetScaler appliance. Each client certificate

is unique and should be used by only one client system.

Whether you obtain the client certificate from a CA, use an existing client certificate, or generate a client certificate on the NetScaler appliance, you must convert the certificate to the correct format. On the NetScaler, certificates are stored in either the PEM or DER format and must be converted to PKCS#12 format before they are installed on the client system. After converting the certificate and transferring it to the client, system, make sure that it is installed on that system and configured for the client application that will be part of the SSL transactions (for example, the web browser).

For instructions on how to convert a certificate from PEM or DER format to PKCS#12 format, see Converting SSL Certificates for Import or Export.

For instructions on how to generate a client certificate, see Generating Self-Signed Certificates.

Enabling Client-Certificate-Based Authentication

Updated: 2013-08-20

By default, client authentication is disabled on the NetScaler appliance, and all SSL transactions proceed without authenticating the client. You can configure client authentication to be either optional or mandatory as part of the SSL handshake.

If client authentication is optional, the NetScaler requests the client certificate but proceeds with the SSL transaction even if the client presents an invalid certificate. If client authentication is mandatory, the NetScaler terminates the SSL handshake if the SSL client does not provide a valid certificate.

Caution: Citrix recommends that you define proper access control policies before changing client-certificate-based authentication check to optional.
Note: Client authentication is configured for individual SSL virtual servers, not globally.

To enable client-certificate-based authentication by using the command line interface

At the command prompt, type the following commands to enable the client-certificate-based authentication and verify the configuration:

- set ssl vserver <vServerName> [-clientAuth (ENABLED | DISABLED)] [-clientCert (MANDATORY | OPTIONAL)]
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vssl -clientAuth ENABLED -clientCert Mandatory
 Done
> show ssl vserver vssl

      Advanced SSL configuration for VServer vssl:
      DH: DISABLED
      Ephemeral RSA: ENABLED        Refresh Count: 0
      Session Reuse: ENABLED        Timeout: 120 seconds
      Cipher Redirect: DISABLED
      SSLv2 Redirect: DISABLED
      ClearText Port: 0
```



```
    Client Auth: ENABLED    Client Cert Required: Mandatory
    SSL Redirect: DISABLED
    Non FIPS Ciphers: DISABLED
    SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)    CertKey Name: sslckey   Server Certificate

1)    Policy Name: client_cert_policy  Priority: 0

1)    Cipher Name: DEFAULT
    Description: Predefined Cipher Alias
 Done
```

To enable client-certificate-based authentication by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server for which you want to configure client certificate-based authentication, and then click Open.
3. Click the SSL Settings tab, and then click SSL Parameters.
4. In the Others group, select the Client Authentication check box.
5. In Client Certificate, select Mandatory.

   Note: To configure optional client authentication in Client Certificate, click Optional.
6. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK. The virtual server is now configured for client authentication.

Binding CA Certificates to the Virtual Server

A CA whose certificate is present on the NetScaler appliance must issue the client certificate used for client authentication. You must bind this certificate to the NetScaler virtual server that will carry out client authentication.

You must bind the CA certificate to the SSL virtual server in such a way that the NetScaler can form a complete certificate chain when it verifies the client certificate. Otherwise, certificate chain formation fails and the client is denied access even if its certificate is valid.

You can bind CA certificates to the SSL virtual server in any order. The NetScaler forms the proper order during client certificate verification.

For example, if the client presents a certificate issued by **CA_A**, where **CA_A** is an intermediate CA whose certificate is issued by **CA_B**, whose certificate is in turn issued by a trusted root CA, **Root_CA**, a chain of certificates that contain all three of these certificates must be bound to the virtual server on the NetScaler.

For instructions on binding one or more certificates to the virtual server, see Binding the Certificate-key Pair to the SSL Based Virtual Server.

For instructions on creating a chain of certificates, see Creating a Chain of Certificates.

Customizing the SSL Configuration

Jun 03, 2015

Once your basic SSL configuration is operational, you can customize some of the parameters that are specific to the certificates being used in SSL transactions. You can also enable and disable session reuse and client authentication, and you can configure redirect responses for cipher and SSLv2 protocol mismatches.

You can also customize SSL settings for two NetScaler appliances in a High Availability configuration, and you can synchronize settings, certificates and keys across those appliances.

These settings will depend on your network deployment and the type of clients you expect will connect to your servers.

To customize the SSL configuration, see the following sections:

- Configuring Diffie-Hellman (DH) Parameters
- Configuring Ephemeral RSA
- Configuring Session Reuse
- Configuring Cipher Redirection
- Configuring SSLv2 Redirection
- Configuring SSL Protocol Settings
- Configuring Close-Notify
- Configuring ECDHE Ciphers
- Configuring a Common Name on an SSL Service or Service Group for Server Certificate Authentication
- Configuring Advanced SSL Settings
- Synchronizing Configuration Files in a High Availability Setup
- Managing Server Authentication
- Configuring User-Defined Cipher Groups on the NetScaler Appliance

Configuring Diffie-Hellman (DH) Parameters

Dec 22, 2014

If you are using ciphers on the NetScaler that require a DH key exchange to set up the SSL transaction, enable DH key exchange on the NetScaler and configure other settings based on your network.

To list the ciphers for which DH parameters must be set by using the NetScaler command line, type: sh cipher DH.

To list the ciphers for which DH parameters must be set by using the configuration utility, navigate to Traffic Management > SSL > Cipher Groups, and double-click DH.

For details on how to enable DH key exchange, see Generating a Diffie-Hellman (DH) Key.

To configure DH Parameters by using the command line interface

At the command prompt, type the following commands to configure DH parameters and verify the configuration:

- set ssl vserver <vserverName> -dh <Option> -dhCount <RefreshCountValue> -filepath <string>
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vs-server -dh ENABLED  -dhFile /nsconfig/ssl/ns-server.cert -dhCount 1000
 Done
> show ssl vserver vs-server

      Advanced SSL configuration for VServer vs-server:
      DH: ENABLED
      Ephemeral RSA: ENABLED        Refresh Count: 1000
      Session Reuse: ENABLED        Timeout: 120 seconds
      Cipher Redirect: DISABLED
      SSLv2 Redirect: DISABLED
      ClearText Port: 0
      Client Auth: DISABLED
      SSL Redirect: DISABLED
      Non FIPS Ciphers: DISABLED
      SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)    Cipher Name: DEFAULT
      Description: Predefined Cipher Alias
 Done
```

To configure DH Parameters by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server for which you want to customize SSL settings, and then click **Open**.
3. On the SSL Settings tab, click SSL Parameters.
4. In the Configure SSL Params dialog box, specify values for the following parameters:
   - Enable DH Param*

- Refresh Count
- File Path*

* A required parameter

5. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK. The DH parameters are now configured.

Configuring Ephemeral RSA

Aug 20, 2013

Ephemeral RSA allows export clients to communicate with the secure server even if the server certificate does not support export clients (1024-bit certificate). If you want to prevent export clients from accessing the secure web object and/or resource, you need to disable ephemeral RSA key exchange.

By default, this feature is enabled on the NetScaler appliance, with the refresh count set to zero (infinite use).

Note:
The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted but reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the system restarts.

To configure Ephemeral RSA by using the command line interface

At the command prompt, type the following commands to configure ephemeral RSA and verify the configuration:

- set ssl vserver <vServerName> -eRSA (enabled | disabled) -eRSACount <positive_integer>
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vs-server -eRSA ENABLED -eRSACount 1000
 Done
> show ssl vserver vs-server

      Advanced SSL configuration for VServer vs-server:
      DH: DISABLED
      Ephemeral RSA: ENABLED        Refresh Count: 1000
      Session Reuse: ENABLED        Timeout: 120 seconds
      Cipher Redirect: DISABLED
      SSLv2 Redirect: DISABLED
      ClearText Port: 0
      Client Auth: DISABLED
      SSL Redirect: DISABLED
      Non FIPS Ciphers: DISABLED
      SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)     Cipher Name: DEFAULT
      Description: Predefined Cipher Alias
 Done
```

To configure Ephemeral RSA by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server for which you want to customize SSL settings, and then click Open.
3. On the SSL Settings tab, click SSL Parameters.

4. In the Configure SSL Params dialog box, specify values for the following parameters:
   - Enable Ephemeral RSA*
   - Refresh Count*

   * A required parameter
5. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK. The ephemeral RSA parameters are now configured.

Configuring Session Reuse

Aug 20, 2013

For SSL transactions, establishing the initial SSL handshake requires CPU-intensive public key encryption operations. Most handshake operations are associated with the exchange of the SSL session key (client key exchange message). When a client session is idle for some time and is then resumed, the SSL handshake is typically conducted all over again. With session reuse enabled, session key exchange is avoided for session resumption requests received from the client.

Session reuse is enabled on the NetScaler appliance by default. Enabling this feature reduces server load, improves response time, and increases the number of SSL transactions per second (TPS) that can be supported by the server.

To configure session reuse by using the command line interface

At the command prompt, type the following commands to configure session reuse and verify the configuration:

- set ssl vserver <vServerName> -sessReuse (ENABLED | DISABLED) -sessTimeout <positive_integer>
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vs-ssl -sessreuse enabled -sesstimeout 600
 Done

> show ssl vserver vs-ssl

        Advanced SSL configuration for VServer vs-ssl:
        DH: DISABLED
        Ephemeral RSA: ENABLED        Refresh Count: 1000
        Session Reuse: ENABLED        Timeout: 600 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: DISABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)      CertKey Name: Auth-Cert-1       Server Certificate

1)      Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
 Done
```

To configure session reuse by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server for which you want to customize SSL settings, and then click Open.
3. On the SSL Settings tab, click SSL Parameters.
4. In the Configure SSL Params dialog box, specify values for the following parameters:

- Enable Session Reuse*
- Time-out

* A required parameter

5. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK.

Configuring Cipher Redirection

Aug 20, 2013

During the SSL handshake, the SSL client (usually a web browser) announces the suite of ciphers that it supports, in the configured order of cipher preference. From that list, the SSL server then selects a cipher that matches its own list of configured ciphers.

If the ciphers announced by the client do not match those configured on the SSL server, the SSL handshake fails, and the failure is announced by a cryptic error message displayed in the browser. These messages rarely mention the exact cause of the error.

With cipher redirection, you can configure an SSL virtual server to deliver accurate, meaningful error messages when an SSL handshake fails. When SSL handshake fails, the NetScaler appliance redirects the user to a previously configured URL or, if no URL is configured, displays an internally generated error page.

To configure cipher redirection by using the command line interface

At the command prompt, type the following commands to configure cipher redirection and verify the configuration:

- set ssl vserver <vServerName> -cipherRedirect < ENABLED | DISABLED> -cipherURL < URL>
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vs-ssl -cipherRedirect ENABLED -cipherURL http://redirectURl
 Done
> show ssl vserver vs-ssl

        Advanced SSL configuration for VServer vs-ssl:
        DH: DISABLED
        Ephemeral RSA: ENABLED        Refresh Count: 1000
        Session Reuse: ENABLED        Timeout: 600 seconds
        Cipher Redirect: ENABLED      Redirect URL: http://redirectURl
        SSLv2 Redirect: DISABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)      CertKey Name: Auth-Cert-1       Server Certificate

1)      Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
 Done
```

To configure cipher redirection by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.

2. Select the virtual server for which you want to customize SSL settings, and then click Open.
3. On the SSL Settings tab, click SSL Parameters.
4. In the Configure SSL Params dialog box, specify values for the following parameters:
   - Enable Cipher Redirect
   - Redirect URL
5. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK. The NetScaler is now configured to redirect clients in case of a cipher suite mismatch.

Configuring SSLv2 Redirection

Aug 20, 2013

For an SSL transaction to be initiated, and for successful completion of the SSL handshake, the server and the client should agree on an SSL protocol that both of them support. If the SSL protocol version supported by the client is not acceptable to the server, the server does not go ahead with the transaction, and an error message is displayed.

You can configure the server to display a precise error message (user-configured or internally generated) advising the client on the next action to be taken. Configuring the server to display this message requires that you set up SSLv2 redirection.

To configure SSLv2 redirection by using the command line interface

At the command prompt, type the following commands to configure SSLv2 redirection and verify the configuration:

- set ssl vserver <vServerName> [-sslv2Redirect (ENABLED | DISABLED) [-sslv2URL <URL>]]
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vs-ssl -sslv2Redirect ENABLED -sslv2URL http://sslv2URL
 Done
> show ssl vserver vs-ssl

        Advanced SSL configuration for VServer vs-ssl:
        DH: DISABLED
        Ephemeral RSA: ENABLED        Refresh Count: 1000
        Session Reuse: ENABLED        Timeout: 600 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: ENABLED Redirect URL: http://sslv2URL
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)      CertKey Name: Auth-Cert-1       Server Certificate

1)      Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
 Done
```

To configure SSLv2 redirection by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server for which you want to customize SSL settings, and then click Open.
3. On the SSL Settings tab, click SSL Parameters.
4. In the Configure SSL Params dialog box, specify values for the following parameters:
   - Enable SSLv2 Redirect
   - SSLv2 URL

5. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK. The NetScaler is now configured to redirect clients that only support SSLv2 protocol.

Configuring SSL Protocol Settings

May 20, 2015

The NetScaler appliance supports the SSLv2, SSLv3, TLSv1, TLSv1.1, and TLSv1.2 protocols. Each of these can be set on the appliance as required by your deployment and the type of clients that will connect to the appliance.

Note: Support for TLS protocol versions 1.1 and 1.2 is not available on a FIPS appliance or on a NetScaler virtual appliance.

To configure SSL protocol support by using the command line interface

At the command prompt, type the following commands to configure SSL protocol support and verify the configuration:

- set ssl vserver <vServerName> -ssl2 (ENABLED | DISABLED) -ssl3 (ENABLED | DISABLED) -tls1 (ENABLED | DISABLED) -tls11 (ENABLED | DISABLED) -tls12 (ENABLED | DISABLED)
- show ssl vserver <vServerName>

Example

```
> set ssl vserver vs-ssl -tls11 ENABLED -tls12 ENABLED
 Done
> sh ssl vs vs-ssl

        Advanced SSL configuration for VServer vs-ssl:
        DH: DISABLED
        Ephemeral RSA: ENABLED                  Refresh Count: 0
        Session Reuse: ENABLED                   Timeout: 120 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: DISABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SNI: DISABLED
        SSLv2: DISABLED      SSLv3: ENABLED    TLSv1.0: ENABLED  TLSv1.1: ENABLED  TLSv1.2: ENABLED
        Push Encryption Trigger: Always
        Send Close-Notify: YES

        1 bound certificate:
1)       CertKey Name: mycert  Server Certificate

        1 configured cipher:
1)       Cipher Name: DEFAULT
        Description: Predefined Cipher Alias
Done
```

To configure SSL protocol support by using the configuration Utility

1. In the navigation pane, expand SSL Offload, and then click Virtual Servers.
2. Navigate to Traffic Management > SSL Offload > Virtual Servers.
3. Select the virtual server for which you want to customize SSL settings, and then click Open.
4. On the SSL Settings tab, click SSL Parameters.

5. In the Configure SSL Params dialog box, in the SSL Protocol group, select any of the following protocol options that you want to enable:
   - TLSv1.2
   - TLSv1.1
   - TLSv1
   - SSLv3
   - SSLv2
6. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK.

Configuring Close-Notify

Aug 20, 2013

A close-notify is a secure message that indicates the end of SSL data transmission. A close-notify setting is required at the global level. This setting applies to all virtual servers, services, and service groups. For information about the global setting, see Configuring Advanced SSL Settings.

In addition to the global setting, you can set the close-notify parameter at the virtual server, service, or service group level. You therefore have the flexibility of setting the parameter for one entity and unsetting it for another entity. However, make sure that you set this parameter at the global level. Otherwise, the setting at the entity level does not apply.

To configure close-notify at the entity level by using the command line interface

At the command prompt, type any of the following commands to configure close-notify and verify the configuration:

1. To configure close-notify at the virtual server level, type:
   - set ssl vserver <vServerName> -sendCloseNotify (**YES** | **NO**)
   - show ssl vserver <vServerName>
2. To configure close-notify at the service level, type:
   - set ssl service <serviceName> -sendCloseNotify (**YES** | **NO**)
   - show ssl service <serviceName>
3. To configure close-notify at the service group level, type:
   - set ssl serviceGroup <serviceGroupName> -sendCloseNotify (**YES** | **NO**)
   - show ssl serviceGroup <serviceGroupName>

Example

```
> set ssl vserver sslvsvr -sendCloseNotify YES
 Done
```

To configure close-notify at the entity level by using the configuration utility

1. Navigate to Traffic Management > SSL Offload.
2. Click Virtual Servers or Services or Service Groups.
3. Select the virtual server, service, or service group for which you want to customize SSL settings, and then click Open.
4. On the SSL Settings tab, click SSL Parameter.
5. In the Configure SSL Params dialog box, select Send Close-Notify.
6. Click OK, and in the Configure Virtual Server (SSL Offload) dialog box, click OK.

/

AppDNA

Citrix Cloud

Citrix Receiver

CloudBridge

CloudPortal Services Manager

NetScaler

NetScaler Gateway

NetScaler SD-WAN

ShareFile

VDI-in-a-Box

XenApp and XenDesktop

XenMobile

XenServer

Advanced Concepts

Developer

Legacy Documentation

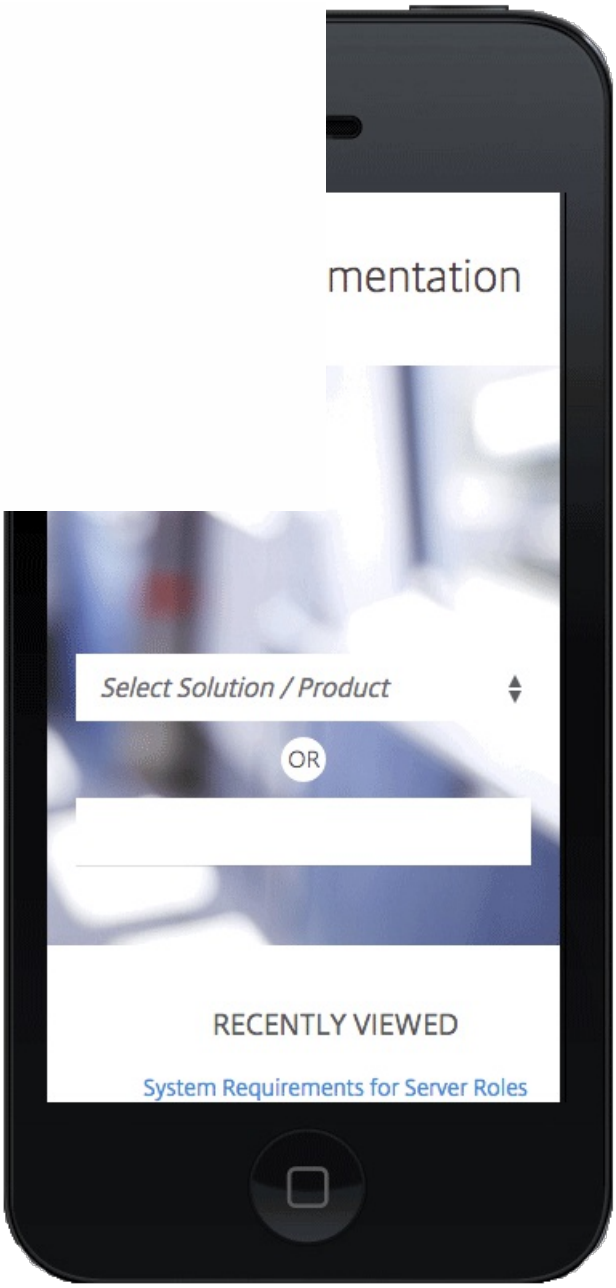
e feel your pain.

' is not here. The link might be misspelled or outdated.

arch or navigate for the content
d retry the link

vestigate

Feedback link at the bottom of Docs.citrix.com to tell us about it



Configuring a Common Name on an SSL Service or Service Group for Server Certificate Authentication

May 13, 2015

In end-to-end encryption with server authentication enabled, you can include a common name in the configuration of an SSL service or service group. The name that you specify is compared to the common name in the server certificate during an SSL handshake. If the two names match, the handshake is successful. This configuration is especially useful if there are, for example, two servers behind a firewall and one of the servers spoofs the identity of the other. If the common name is not checked, a certificate presented by either server is accepted if the IP address matches.

To configure common-name verification for an SSL service or service group by using the command line interface

At the command prompt, type the following commands to specify server authentication with common-name verification and verify the configuration:

1. To configure common name in a service, type:
   - set ssl service <serviceName> -commonName <string> -serverAuth ENABLED
   - show ssl service <serviceName>
2. To configure common name in a service group, type:
   - set ssl serviceGroup <serviceGroupName> -commonName <string> -serverAuth ENABLED
   - show ssl serviceGroup <serviceGroupName>

Example

```
> set ssl service svc1 -commonName xyz.com -serverAuth ENABLED
 Done
> show ssl service svc1
 Advanced SSL configuration for Back-end SSL Service svc1:
 DH: DISABLED
 Ephemeral RSA: DISABLED
 Session Reuse: ENABLED  Timeout: 300 seconds
 Cipher Redirect: DISABLED
 SSLv2 Redirect: DISABLED
 Server Auth: ENABLED Common Name: www.xyz.com
 SSL Redirect: DISABLED
 Non FIPS Ciphers: DISABLED
 SNI: DISABLED
 SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED

1) CertKey Name: cacert CA Certificate  OCSPCheck: Optional

1) Cipher Name: ALL
 Description: Predefined Cipher Alias
 Done
```

To configure common-name verification for an SSL service or service group by using the configuration utility

1. Navigate to Traffic Management > SSL Offload.
2. Click Services or Service Groups.
3. Select the service or service group for which you want to customize SSL settings, and then click Open.
4. On the SSL Settings tab, click SSL Parameters.
5. In the Configure SSL Params dialog box, select Server Authentication, and then in the Common Name field, specify a name for the certificate.
6. Click OK, and in the Configure Service dialog box, click OK.

Configuring Advanced SSL Settings

Sep 20, 2013

Advanced customization of your SSL configuration addresses specific issues. You can use the set ssl parameter command or the configuration utility to specify the following:

- Quantum size to be used for SSL transactions.
- CRL memory size.
- OCSP cache size.
- Deny SSL renegotiation.
- Set the PUSH flag for decrypted, encrypted, or all records.
- Drop requests if the client initiates the handshake for one domain and sends an HTTP request for another domain.
- Set the time after which encryption is triggered.
  Note: The time that you specify applies only if you use the set ssl vserver command or the configuration utility to set timer-based encryption.

To configure advanced SSL settings by using the command line interface

At the command prompt, type the following commands to configure advanced SSL settings and verify the configuration:

- set ssl parameter [-quantumSize <quantumSize>] [-crlMemorySizeMB <positive_integer>] [-strictCAChecks (YES | NO)] [-sslTriggerTimeout <positive_integer>] [-sendCloseNotify (YES | NO)] [-encryptTriggerPktCount <positive_integer>] [-denySSLReneg <denySSLReneg>] [-insertionEncoding (Unicode | UTF-8)] [-ocspCacheSize <positive_integer>][- pushFlag <positive_integer>] [- dropReqWithNoHostHeader (YES | NO)] [-pushEncTriggerTimeout <positive_integer>]
- show ssl parameter

Example

```
> set ssl parameter -quantumSize 8 -crlMemorySizeMB 256 -strictCAChecks no -sslt
iggerTimeout 100 -sendClosenotify no -encryptTriggerPktCount 45 -denySSLReneg no
-insertionEncoding unicode -ocspCacheSize 10 -pushFlag 3 -dropReqWithNoHostHeader YES  -pushEncTriggerTimeout 100 ms
 Done

> show ssl parameter
Advanced SSL Parameters
-----------------------
      SSL quantum size:            8 kB
      Max CRL memory size:         256 MB
      Strict CA checks:            NO
      Encryption trigger timeout     100 mS
      Send Close-Notify            NO
      Encryption trigger packet count:      45
      Deny SSL Renegotiation       NO
      Subject/Issuer Name Insertion Format:   Unicode
      OCSP cache size:      10 MB
      Push flag:     0x3 (On every decrypted and encrypted record)
      Strict Host Header check for SNI enabled SSL sessions:        YES
       PUSH encryption trigger timeout 100 ms
 Done
```

To configure advanced SSL settings by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. In the details pane, under Settings, click Change advanced SSL settings.
3. In the Change advanced SSL settings dialog box, set the following parameters:
   - SSL quantum size (Kbytes)
   - Max CRL memory size (Mbytes)
   - Encryption trigger timeout (10 mS ticks)
   - Encryption trigger packet count

- Deny SSL Renegotiation
- OCSP cache size(Mbytes)
- Encoding type
- PUSH encryption trigger timeout (msec)
- Strict CA checks
- Send Close-Notify
- PUSH Flag Insertion
- Drop requests for SNI enabled SSL sessions if Host header is absent

4. Click OK. The parameters you selected are now enabled on the appliance.

PUSH Flag-Based Encryption Trigger Mechanism

The encryption trigger mechanism that is based on the PSH TCP flag now enables you to do the following:

- Merge consecutive packets in which the PSH flag is set into a single SSL record, or ignore the PSH flag.
- Perform timer-based encryption, in which the time-out value is set globally by using the set ssl parameter -pushEncTriggerTimeout <positive_integer> command.

To configure PUSH flag-based encryption by using the command line interface

At the command prompt, type the following commands to configure PUSH flag-based encryption and verify the configuration:

- set ssl vserver <vServerName> [-pushEncTrigger <pushEncTrigger>]
- show ssl vserver

Example

```
Advanced SSL configuration for VServer v1:
        DH: DISABLED
        Ephemeral RSA: ENABLED                  Refresh Count: 0
        Session Reuse: ENABLED                  Timeout: 120 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: DISABLED
        ClearText Port: 0
        Client Auth: DISABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SNI: DISABLED
        SSLv2: DISABLED         SSLv3: ENABLED          TLSv1: ENABLED
        Push Encryption Trigger: Always
```

To configure PUSH flag-based encryption by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. Select the virtual server for which you want to customize PUSH-flag based encryption, and then click Open.
3. On the SSL Settings tab, click SSL Parameters.
4. In the Configure SSL Params dialog box, select a value for the PUSH Encryption Trigger parameter.
5. Click OK and, in the Configure Virtual Server (SSL Offload) dialog box, again click OK. The PSH flag-based encryption trigger is now configured.

Synchronizing Configuration Files in a High Availability Setup

Oct 28, 2013

In a high availability (HA) set up, the primary NetScaler appliance in the HA pair automatically synchronizes with the secondary appliance in the pair. In the synchronization process, the secondary copies the primary's /nsconfig/ssl/ directory, which is the default location for storing the certificates and keys for SSL transactions. Synchronization occurs at one-minute intervals and every time a new file is added to the directory.

To synchronize files in a high availability setup by using the command line interface

At the command prompt, type the following command:

sync HA files [<Mode>]

Example

```
sync HA files SSL
```

To synchronize files in a high availability setup by using the configuration utility

1. Navigate to Traffic Management > SSL.
2. In the details pane, under Tools, click Start file synchronization.
3. In the Start file synchronization dialog box, in the Mode drop-down list, select the appropriate type of synchronization (for example, SSL certificates and Keys), and then click OK.

Managing Server Authentication

Aug 20, 2013

Since the NetScaler appliance performs SSL offload and acceleration on behalf of a web server, the appliance does not usually authenticate the Web server's certificate. However, you can authenticate the server in deployments that require end-to-end SSL encryption.

In such a situation, the NetScaler becomes the SSL client, carries out a secure transaction with the SSL server, verifies that a CA whose certificate is bound to the SSL service has signed the server certificate, and checks the validity of the server certificate.

To authenticate the server, you must first enable server authentication and then bind the certificate of the CA that signed the server's certificate to the SSL service on the NetScaler. When binding the certificate, you must specify the bind as CA option.

To enable (or disable) server certificate authentication by using the command line interface

At the command prompt, type the following commands to enable server certificate authentication and verify the configuration:

- set ssl service <serviceName> -serverAuth (ENABLED | DISABLED)
- show ssl service <serviceName>

Example

```
> set ssl service ssl-service-1 -serverAuth ENABLED
 Done
> show ssl service ssl-service-1

      Advanced SSL configuration for Back-end SSL Service ssl-service-1:
      DH: DISABLED
      Ephemeral RSA: DISABLED
      Session Reuse: ENABLED        Timeout: 300 seconds
      Cipher Redirect: DISABLED
      SSLv2 Redirect: DISABLED
      Server Auth: ENABLED
      SSL Redirect: DISABLED
      Non FIPS Ciphers: DISABLED
      SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)     Cipher Name: ALL
      Description: Predefined Cipher Alias
 Done
```

To enable (or disable) server certificate authentication by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Services.
2. Select the service for which you want to enable server authentication, and then click Open.
3. In Configure Service dialog box, on the SSL Settings tab, click SSL Parameters.

4. In the Others group, select Server Authentication.
5. Click OK. Server authentication is now enabled for the service.

To bind the CA certificate to the service by using the command line interface

At the command prompt, type the following commands to bind the CA certificate to the service and verify the configuration:

- bind ssl service <serviceName> -certkeyName <string> -CA
- show ssl service <serviceName>

Example

```
> bind ssl service ssl-service-1 -certkeyName samplecertkey -CA
 Done
> show ssl service ssl-service-1

        Advanced SSL configuration for Back-end SSL Service ssl-service-1:
        DH: DISABLED
        Ephemeral RSA: DISABLED
        Session Reuse: ENABLED        Timeout: 300 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: DISABLED
        Server Auth: ENABLED
        SSL Redirect: DISABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)     CertKey Name: samplecertkey     CA Certificate        CRLCheck: Optional

1)     Cipher Name: ALL
       Description: Predefined Cipher Alias
 Done
```

Configuring User-Defined Cipher Groups on the NetScaler Appliance

Oct 02, 2013

A cipher group is a set of cipher suites that you bind to an SSL virtual server, service, or service group on the NetScaler appliance. A cipher suite comprises a protocol, a key exchange (Kx) algorithm, an authentication (Au) algorithm, an encryption (Enc) algorithm, and a message authentication code (Mac) algorithm. Your appliance ships with a predefined set of cipher groups. When you create a SSL service or SSL service group, the ALL cipher group is automatically bound to it. However, when you create an SSL virtual server or a transparent SSL service, the DEFAULT cipher group is automatically bound to it. In addition, you can create a user-defined cipher group and bind it to an SSL virtual server, service, or service group.

Note: If your MPX appliance does not have any licenses, then only the EXPORT cipher is bound to your SSL virtual server, service, or service group.

To create a user-defined cipher group, first you create a cipher group and then you bind ciphers or cipher groups to this group. If you specify a cipher alias or a cipher group, all the ciphers in the cipher alias or group are added to the user-defined cipher group. You can also add individual ciphers (cipher suites) to a user-defined group. However, you cannot modify a predefined cipher group. Before removing a cipher group, unbind all the cipher suites in the group.

If you bind a cipher group to an SSL virtual server, service, or service group, the ciphers are appended to the existing ciphers that are bound to the entity. To bind a specific cipher group to the entity, you must first unbind the ciphers or cipher group that is bound to the entity and then bind the specific cipher group. For example, to bind only the AES cipher group to an SSL service, you perform the following steps:

1. Unbind the default cipher group ALL that is bound by default to the service when the service is created.
   `unbind ssl service <service name> -cipherName ALL`
2. Bind the AES cipher group to the service
   `bind ssl service <Service name> -cipherName AE`

If you want to bind the cipher group DES in addition to AES, at the command prompt, type:

- `bind ssl service <service name> -cipherName DES`

Note: The free NetScaler virtual appliance supports only the DH cipher group.

To configure a user-defined cipher group by using the command line interface

At the command prompt, type the following commands to add a cipher group, or to add ciphers to a previously created group, and verify the settings:

- add ssl cipher <cipherGroupName>
- bind ssl cipher <cipherGroupName> -cipherName <string>
- show ssl cipher <cipherGroupName>

Example

```
 > add ssl cipher test
 Done
> bind ssl cipher test -cipherName SSLv2
 Done
> show ssl cipher test
```

```
1)     Cipher Name: SSL2-RC2-CBC-MD5
Description: SSLv2 Kx=RSA     Au=RSA  Enc=RC2(128)  Mac=MD5
2)     Cipher Name: SSL2-RC4-MD5
Description: SSLv2 Kx=RSA     Au=RSA  Enc=RC4(128)  Mac=MD5
3)     Cipher Name: SSL2-DES-CBC3-MD5
Description: SSLv2 Kx=RSA     Au=RSA  Enc=3DES(168) Mac=MD5
4)     Cipher Name: SSL2-DES-CBC-MD5
Description: SSLv2 Kx=RSA     Au=RSA  Enc=DES(56)   Mac=MD5
5)     Cipher Name: SSL2-RC4-64-MD5
Description: SSLv2 Kx=RSA     Au=RSA  Enc=RC4(64)   Mac=MD5
6)     Cipher Name: SSL2-EXP-RC4-MD5
Description: SSLv2 Kx=RSA(512) Au=RSA  Enc=RC4(40)   Mac=MD5  Export
7)     Cipher Name: SSL2-EXP-RC2-CBC-MD5
Description: SSLv2 Kx=RSA(512) Au=RSA  Enc=RC2(40)   Mac=MD5  Export
 Done
```

To unbind ciphers from a cipher group by using the command line interface

At the command prompt, type the following commands to unbind ciphers from a user-defined cipher group, and verify the settings:

- show ssl cipher <cipherGroupName>
- unbind ssl cipher <cipherGroupName> -cipherName <string>
- show ssl cipher <cipherGroupName>

Example

```
> show ssl cipher test
1) Cipher Name: SSL2-RC2-CBC-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=RC2(128) Mac=MD5
2) Cipher Name: SSL2-RC4-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=RC4(128) Mac=MD5
3) Cipher Name: SSL2-DES-CBC3-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=3DES(168) Mac=MD5
4) Cipher Name: SSL2-DES-CBC-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=DES(56) Mac=MD5
5) Cipher Name: SSL2-RC4-64-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=RC4(64) Mac=MD5
6) Cipher Name: SSL2-EXP-RC4-MD5
Description: SSLv2 Kx=RSA(512) Au=RSA Enc=RC4(40) Mac=MD5 Export
7) Cipher Name: SSL2-EXP-RC2-CBC-MD5
Description: SSLv2 Kx=RSA(512) Au=RSA Enc=RC2(40) Mac=MD5 Export
Done

> unbind ssl cipher test -cipherName SSL2-RC2-CBC-MD5

> show ssl cipher test
1) Cipher Name: SSL2-RC4-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=RC4(128) Mac=MD5
2) Cipher Name: SSL2-DES-CBC3-MD5
```

```
Description: SSLv2 Kx=RSA Au=RSA Enc=3DES(168) Mac=MD5
3) Cipher Name: SSL2-DES-CBC-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=DES(56) Mac=MD5
4) Cipher Name: SSL2-RC4-64-MD5
Description: SSLv2 Kx=RSA Au=RSA Enc=RC4(64) Mac=MD5
5) Cipher Name: SSL2-EXP-RC4-MD5
Description: SSLv2 Kx=RSA(512) Au=RSA Enc=RC4(40) Mac=MD5 Export
6) Cipher Name: SSL2-EXP-RC2-CBC-MD5
Description: SSLv2 Kx=RSA(512) Au=RSA Enc=RC2(40) Mac=MD5 Export
Done
```

To remove a cipher group by using the command line interface

Note: You cannot remove a built-in cipher group. Before removing a user-defined cipher group, make sure that the cipher group is empty.
At the command prompt, type the following commands to remove a user-defined cipher group, and verify the configuration:

- rm ssl cipher <userDefCipherGroupName> [<cipherName> ...]
- show ssl cipher <cipherGroupName>

Example

```
> rm ssl cipher test
Done

> sh ssl cipher test
ERROR: No such resource [cipherGroupName, test]
```

To configure a user-defined cipher group by using the configuration utility

1. Navigate to Traffic Management > SSL > Cipher Groups.
2. In the details pane, do one of the following:
   - To create a new cipher group, click Add.
   - To modify an existing cipher group, select the cipher group, and then click Open.
3. If creating a new cipher group, in the Create Cipher Group dialog box, in the Cipher Group Name box, type a name for the new cipher group.
4. In the Create Cipher Group or View Cipher Group dialog box, do any or all of the following:
   - Select a cipher group or alias in the Available Cipher Groups list and click Add to move the group to the Configured Cipher Groups list.
   - Select a cipher group or alias in the Available Cipher Groups list, then select ciphers from the Available Ciphers list, and then click Add to move the selected ciphers to the Configured Ciphers list.
   - To move a cipher group or cipher from the Configured list to the Available list, select the group or cipher and click Remove.
5. Click Create, and then click Close. If you created a new cipher group, it appears in the Cipher Groups pane.

To bind a cipher group to an SSL virtual server, service, or service group by using the command line interface

At the command prompt, type one of the following:

- bind ssl vserver <vServerName> -cipherName <string>
- bind ssl service <serviceName> -cipherName <string>
- bind ssl serviceGroup <serviceGroupName> -cipherName <string>

Examples

```
> bind ssl vserver ssl_vserver_test -cipherName test
Done
bind ssl service  nshttps -cipherName test
Done
> bind ssl servicegroup  ssl_svc  -cipherName test
Done
```

To bind a cipher group to an SSL virtual server, service, or service group by using the configuration utility

1. Navigate to Traffic Management > SSL Offload > Virtual Servers or Traffic Management > SSL Offload > Services or Traffic Management > SSL Offload > Service Groups.
2. In the details pane, select the virtual server, service, or service group to bind the cipher to, and then click Open.
3. In the Configure Virtual Server (SSL Offload), Configure Service, or Configure Service Group dialog box, on the SSL Settings tab, click Ciphers.
4. In the SSL-Offload - Configure Ciphers, Service - Configure Ciphers, or Service Group - Configure Ciphers dialog box, do one or both of the following:
   - To bind a cipher group, select a cipher group or alias from the Available Cipher Groups list, and then click Add. To unbind a group, select the cipher group or alias from the Configured Cipher Groups list, and then click Remove.
   - To bind a cipher, select a cipher group or alias from the Available Cipher Groups list, then select ciphers from the Available Ciphers list, and then click Add. To unbind a cipher. select the cipher from the Configured Ciphers list, and then click Remove.

   Note: To override an existing cipher or cipher group, drag and drop the configured cipher or cipher group to a new location in the Configured Ciphers list or the Configured Cipher Groups list so that it precedes the cipher or cipher group to be overridden.
5. Click OK to close the dialog box, and then click OK again.

Configuring SSL Actions and Policies

Jun 03, 2015

An SSL policy evaluates incoming traffic and applies a predefined action to requests that match a rule (expression). You have to configure the actions before creating the policies, so that you can specify an action when you create a policy. To put a policy into effect, you must either bind it to a virtual server on the appliance, so that it applies only to traffic flowing through that virtual server, or bind it globally, so that it applies to all traffic flowing through the appliance.

SSL actions define SSL settings that you can apply to the selected requests. You associate an action with one or more policies. Data in client connection requests or responses is compared to a rule specified in the policy, and the action is applied to connections that match the rule (expression).

You can configure classic policies with classic expressions and default syntax policies with default syntax expressions for SSL.
Note: Users who are not experienced in configuring policies at the NetScaler command line usually find using the configuration utility to be considerably easier.
You can associate a user-defined action or a built-in action to a default syntax policy. Classic policies allow only user-defined actions. In default syntax policy, you can also group policies under a policy label, in which case they are applied only when invoked from another policy.

Common uses of SSL actions and policies include per-directory client authentication, support for Outlook web access, and SSL-based header insertions. SSL-based header insertions contain SSL settings required by a server whose SSL processing has been offloaded to the NetScaler appliance.

To configure SSL actions and policies, see the following sections:

- Configuring User-Defined Actions for SSL Policies
- Configuring SSL Policies
- Configuring an SSL Default Syntax Policy
- Configuring Built-in Actions for SSL Default Syntax Policies
- Configuring SSL Policy Labels
- Configuring Per-Directory Client Authentication
- Configuring Support for Outlook Web Access
- Configuring SSL-Based Header Insertion
- Binding SSL Policies to a Virtual Server
- Binding SSL Policies Globally

Configuring User-Defined Actions for SSL Policies

Sep 05, 2013

SSL policies require that you create an action before creating a policy, so that you can specify the actions when you create the policies. In SSL default syntax policies, you can also use the built-in actions. For more information about built-in actions, see Configuring Built-in SSL Actions.

At the command prompt, type the following commands to configure an action and verify the configuration:

- add SSL action <name> -clientAuth(DOCLIENTAUTH | NOCLIENTAUTH) -clientCert (ENABLED | DISABLED) certHeader <string> -clientHeader <string> -clientCertSerialNumber (ENABLED | DISABLED) -certSerialHeader <string> **-clientCertSubject** (ENABLED | DISABLED) -certSubjectHeader <string> -clientCertHash (ENABLED | DISABLED) -certHashHeader <string> -clientCertIssuer (ENABLED | DISABLED) -certIssuerHeader <string> -sessionID (ENABLED | DISABLED) -sessionIDheader <string> -cipher (ENABLED | DISABLED) -cipherHeader <string> -clientCertNotBefore (ENABLED | DISABLED) **-certNotBeforeHeader** <string> -clientCertNotAfter (ENABLED | DISABLED) -certNotAfterHeader <string> -OWASupport (ENABLED | DISABLED)
- show ssl action [<name>]

Example

```
> add ssl action Action-SSL-ClientCert -clientCert ENABLED -certHeader "X-Client-Cert"
 Done
> show ssl action Action-SSL-ClientCert
1)     Name: Action-SSL-ClientCert
       Data Insertion Action:
      Cert Header: ENABLED          Cert Tag: X-Client-Cert
 Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. On the Actions tab, in the details pane, click Add.
3. In the Create SSL Action dialog box, set the following parameters:
   - Name*
   - Client Authentication
   - Client Certificate
   - Certificate Tag
   - Client Certificate Serial Number
   - Serial Number Tag
   - Client Certificate Subject (DN)
   - Subject Tag
   - Client Certificate Hash
   - Hash Tag
   - Client Certificate Issuer
   - Issuer Tag
   - Session ID
   - Session ID Tag

- Cipher Suite
- Cipher Tag
- Client Certificate Not Before Date
- Not Before Tag
- Client Certificate Not After Date
- Not After Tag
- Outlook Web Access

* A required parameter

4. Click Create, and then click Close.

Configuring SSL Policies

Sep 26, 2013

Policies on the NetScaler help identify specific connections that you want to process. The processing is based on the actions that are configured for that particular policy. Once you create the policy and configure an action for it, you must either bind it to a virtual server on the NetScaler, so that it applies only to traffic flowing through that virtual server, or bind it globally, so that it applies to all traffic flowing through any virtual server configured on the NetScaler.

The NetScaler SSL feature supports both classic policies and default syntax policies . For a complete description of classic and default syntax expressions, how they work, and how to configure them manually, see Policy Configuration and Reference.

Note: Users who are not experienced in configuring policies at the NetScaler command line will usually find using the configuration utility considerably easier.

Configuring an SSL Default Syntax Policy

Sep 25, 2014

An SSL default syntax policy defines a control or a data action to be performed on requests. SSL policies can therefore be categorized as control policies and data policies:

- **Control policy**. A control policy uses a control action, such as forcing client authentication.
  Note: In release 10.5 or later, deny SSL renegotiation (denySSLReneg) is set, by default, to ALL. However, control policies, such as CLIENTAUTH, trigger a renegotiation handshake. If you use such policies, you must set denySSLReneg to NO.
- **Data policy**. A data policy uses a data action, such as inserting some data into the request.

The essential components of a policy are an expression and an action. The expression identifies the requests on which the action is to be performed. SSL policies use the default expression syntax or the classic expression syntax. For information about expressions and how to configure them, see Policy Configuration and Reference.

You can configure a default syntax policy with a built-in action or a user-defined action. You can configure a policy with a built-in action without creating a separate action. However, to configure a policy with a user-defined action, first configure the action and then configure the policy.

You can specify an additional action, called an UNDEF action, to be performed in the event that applying the expression to a request has an undefined result.

At the command prompt, type:

add ssl policy <name> -rule <expression> -Action <string> [-undefAction <string>] [-comment <string>]

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, click Policies, and then click Add.
3. In the Create SSL Policy dialog box, set the following parameters:
   - Name*
   - Request Action*
   - Undefined-Result Action
   - Expression
   - Comments

   * A required parameter
4. Click Create, and then click Close.
5. On the Policies tab, verify that the settings displayed for the policy that you just configured are correct.

Configuring Built-in Actions for SSL Default Syntax Policies

Aug 20, 2013

Unless you need only the built-in actions in your policies, you have to create the actions before creating the policies, so that you can specify the actions when you create the policies. The built-in actions are of two types, control actions and data actions. You use control actions in control policies, and data actions in data policies.

The built-in control actions are:

- CLIENTAUTH—Perform client certificate authentication.
- NOCLIENTAUTH—Do not perform client certificate authentication.

The built-in data actions are:

- RESET—Close the connection by sending a RST packet to the client.
- DROP—Drop all packets from the client. The connection remains open until the client closes it.
- NOOP—Forward the packet without performing any operation on it.

You can create user-defined data actions. For example, if you enable client authentication, you can create an SSL action to insert client-certificate data into the request header before forwarding the request to the web server. For more information about user-defined actions, see Configuring User-Defined SSL Actions.

If a policy evaluation results in an undefined state, an UNDEF action is performed. For either a data policy or a control policy, you can specify RESET, DROP, or NOOP as the UNDEF action. For a control policy, you also have the option of specifying CLIENTAUTH or NOCLIENTAUTH.

In the following example, if the client sends a cipher other than an EXPORT category cipher, the NetScaler appliance requests client authentication. The client has to provide a valid certificate for a successful transaction.

```
add ssl policy pol1 -rule CLIENT.SSL.CIPHER_EXPORTABLE.NOT -reqAction CLIENTAUTH
```

The following examples assume that client authentication is enabled.

If the version in the certificate provided by the user matches the version in the policy, no action is taken and the packet is forwarded:

```
add ssl policy pol1 -rule CLIENT.SSL.CLIENT_CERT.VERSION.EQ(2) -reqAction NOOP
```

If the version in the certificate provided by the user matches the version in the policy, the connection is dropped:

```
add ssl policy pol1 -rule CLIENT.SSL.CLIENT_CERT.VERSION.EQ(2) -reqAction DROP
```

If the version in the certificate provided by the user matches the version in the policy, the connection is reset:

```
add ssl policy pol1 -rule CLIENT.SSL.CLIENT_CERT.VERSION.EQ(2) -reqAction RESET
```

Configuring SSL Policy Labels

Aug 20, 2013

Policy labels are holders for policies. A policy label helps in managing a group of policies, called a policy bank, which can be invoked from another policy. SSL policy labels can be control labels or data labels, depending on the type of policies that are included in the policy label. You can add only data policies in a data policy label and only control policies in a control policy label. To create the policy bank, you bind policies to the label and specify the order of evaluation of each policy relative to others in the bank of policies for the policy label. At the NetScaler command line, you enter two commands to create a policy label and bind policies to the policy label. In the configuration utility, you select options from a dialog box.

At the command prompt, type:

- add ssl policylabel <labelName> -type (CONTROL | DATA)
- bind ssl policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

Example

```
add ssl policylabel cpl1 -type CONTROL
add ssl policylabel dpl1 -type DATA
bind ssl policylabel cpl1 -policyName ctrlpol -priority 1
bind ssl policylabel dpl1 -policyName datapol -priority 1
```

1. Navigate to Traffic Management > SSL > Policy Labels.
2. In the details pane, click Add.
3. In the Create new SSL Policy Label dialog box, set the following parameters:
   - Name*
   - Type*

   * A required parameter
4. Click Insert Policy, and select from the following:
   - **Policy Name**. The name of an existing policy.
   - **New policy**. Invokes the policy creation editor.
5. Click Create, and then click Close.
6. On the Policies tab, verify that the settings displayed for the policy that you just configured are correct.

Configuring Per-Directory Client Authentication

Aug 20, 2013

If you create an action specifying client-side authentication on a per-directory basis, a client identified by a policy associated with the action is not authenticated as part of the initial SSL handshake. Instead, authentication is carried out every time the client wants to access a specific directory on the web server.

For example, if you have multiple divisions in the company, where each division has a folder in which all its files are stored, and you want to know the identity of each client that tries to access files from a particular directory, such as the finance directory, you can enable per-directory client authentication for that directory.

To enable per-directory client authentication, first configure client authentication as an SSL action, and then create a policy that identifies the directory that you want to monitor. When you create the policy, specify your client-authentication action as the action associated with the policy. Then, bind the policy to the SSL virtual server that will receive the SSL traffic.

At the command prompt, type the following commands to create an SSL action to enable to client authentication and verify the configuration:

- add ssl action <name> [-clientAuth (DOCLIENTAUTH | NOCLIENTAUTH)]
- show ssl action [<name>]
- add ssl policy <name> -rule <expression> [-action <string>] [-undefAction <string>] [-comment <string>]
- show ssl policy [<name>]

Example

```
> add ssl action ssl-action-1 -clientAuth DOCLIENTAUTH
 Done
> show ssl action ssl-action-1
1)     Name: ssl-action-1
       Client Authentication Action: DOCLIENTAUTH
       Hits: 0
       Undef Hits: 0
       Action Reference Count: 1
 Done
> add ssl policy ssl-pol-1 -rule 'REQ.HTTP.METHOD==GET' -reqaction ssl-action-1
> sh ssl policy ssl-pol-1
       Name: ssl-pol-1
       Rule: REQ.HTTP.METHOD == GET
       Action: ssl-action-1
       UndefAction: Use Global
       Hits: 0
       Undef Hits: 0
 Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, on the Actions tab, click Add.
3. In the Create SSL Action dialog box, set values for the following parameters:
   - Name*
   - Client Authentication

   * A required parameter
4. Click Create, and then click Close.

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, on the Policies tab, click Add.
3. In the Create SSL Policy dialog box, set values for the following parameters:
   - Name*
   - Request Action*

   * A required parameter
4. Click Add Expression to add an expression.
5. Click Create, and then click Close.
6. Navigate to Traffic Management > SSL Offload > Virtual Servers.
7. In the details pane, select a virtual server, and then click Open.
8. In the Configure Virtual Server (SSL Offload) dialog box, click SSL Policies.
9. In the Bind/Unbind SSL Policies dialog box, click Insert Policy, and from the list select the policy that you want to bind to the virtual server.

Configuring Support for Outlook Web Access

Aug 20, 2013

If your SSL configuration is offloading SSL transactions from an Outlook Web Access (OWA) server, you must insert a special header field, FRONT-END-HTTPS: ON, in all HTTP requests directed to the OWA servers. This is required for the OWA servers to generate URL links as https:// instead of http://.

When you enable support for OWA on the NetScaler, the header is automatically inserted into the specified HTTP traffic, and you do not need to configure a specific header insertion. Use SSL policies to identify all traffic directed to the OWA server.

Note: You can enable Outlook Web Access support for HTTP-based SSL virtual servers and services only. You cannot apply it to TCP-based SSL virtual servers and services.
To enable OWA support, first configure OWA support as an SSL action, and then create a policy that identifies the virtual servers or services for which you want to enable OWA support. When you create the policy, specify your OWA support action as the action associated with the policy. Then, bind the policy to the SSL virtual server that will receive the SSL traffic.

At the command prompt, type the following commands to create an SSL action to enable OWA support and verify the configuration:

- add ssl action <name> -OWASupport (ENABLED | DISABLED)
- show ssl action [<name>]
- add ssl policy <name> -rule <expression> [-action <string>] [-undefAction <string>] [-comment <string>]
- show ssl policy [<name>]

Example

```
> add ssl action ssl-action-2 -OWASupport ENABLED
 Done
> show ssl action ssl-action-2
1)     Name: ssl-action-2
       Type: Data Insertion
       OWA Support: ENABLED
       Hits: 0
       Undef Hits: 0
       Action Reference Count: 1
 Done
> add ssl policy ssl-pol -rule 'REQ.HTTP.METHOD == GET' -reqaction ssl-action-2
Done
> sh ssl policy ssl-pol
      Name: ssl-pol
      Rule: REQ.HTTP.METHOD == GET
      Action: ssl-action-2
      UndefAction: Use Global
```

```
    Hits: 0
    Undef Hits: 0
Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. On the Actions tab, in the details pane, click Add.
3. In the Create SSL Action dialog box, set the following parameters:
   - Name*
   - Outlook Web Access

   * A required parameter
4. Click Create, and then click Close.
   Note: Outlook Web Access support is applicable only for SSL virtual server based configurations and transparent SSL service based configurations and not for SSL configurations with back-end encryption.

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, on the Policies tab, click Add.
3. In the Create SSL Policy dialog box, set values for the following parameters:
   - Name*
   - Request Action*

   * A required parameter
4. Click Add Expression to add an expression.
5. Click Create, and then click Close.
6. Navigate to Traffic Management > SSL Offload > Virtual Servers.
7. In the details pane, select a virtual server, and then click Open.
8. In the Configure Virtual Server (SSL Offload) dialog box, click SSL Policies.
9. In the Bind/Unbind SSL Policies dialog box, click Insert Policy, and from the list select the policy that you want to bind to the virtual server.

Configuring SSL-Based Header Insertion

Aug 20, 2013

Because the NetScaler appliance offloads all SSL-related processing from the servers, the servers receive only HTTP traffic. In some circumstances, the server needs certain SSL information. For example, security audits of recent SSL transactions require the client subject name (contained in an X509 certificate) to be logged on the server.

Such data can be sent to the server by inserting it into the HTTP header as a name-value pair. You can insert the entire client certificate, if required, or only the specific fields from the certificate, such as the subject, serial number, issuer, certificate hash, SSL session ID, cipher suite, or the not-before or not-after date used to determine certificate validity.

You can enable SSL-based insertion for HTTP-based SSL virtual servers and services only. You cannot apply it to TCP-based SSL virtual servers and services. Also, client authentication must be enabled on the SSL virtual server, because the inserted values are taken from the client certificate that is presented to the virtual server for authentication.

To configure SSL-based header insertion, first create an SSL action for each specific set of information to be inserted, and then create policies that identify the connections for which you want to insert the information. As you create each policy, specify the action that you want associated with the policy. Then, bind the policies to the SSL virtual servers that will receive the SSL traffic.

The following example uses default syntax policies. In the following example, a control policy (ctrlpol) is created to perform client authentication if a request is received for the URL /testsite/file5.html. A data policy (datapol) is created to perform an action (act1) if client authentication is successful, and an SSL action (act1) is added to insert the certificate details and issuer's name in the request before forwarding the request. For other URLs, client authentication is disabled. The policies are then bound to an SSL virtual server (ssl_vserver) that receives the SSL traffic.

Example

```
> add ssl action act1 -clientCert ENABLED -certHeader mycert -clientcertissuer ENABLED -certIssuerHeader myissuer
> add ssl policy datapol -rule HTTP.REQ.URL.EQ(\"/testsite/file5.html\") -action act1
> add ssl policy ctrlpol -rule HTTP.REQ.URL.EQ(\"/testsite/file5.html\") -action CLIENTAUTH
> bind ssl vserver ssl_vserver -policyName ctrlpol -priority 1
> bind ssl vserver ssl_vserver -policyName datapol -priority 1
 Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, on the Actions tab, click Add.
3. In the Create SSL Action dialog box, set the following parameters:
   - Name*
   - Client Certificate
   - Certificate Tag
   - Client Certificate Issuer
   - Issuer Tag

   * A required parameter
4. Click Create, and then click Close.
5. On the tab, click Add to create a control policy.
6. In the Create SSL Policy dialog box, set the following parameters:
   - Name*
   - Expression
   - Request Action

* A required parameter

7. Click Create, and then click Close.
8. Create a data policy by repeating steps 5 through 7.
9. In the navigation pane, expand SSL Offload, and then click Virtual Servers.
10. In the details pane, from the list of virtual servers, select the virtual server to which you want to bind the SSL policies, and then click Open.
11. In the Configure Virtual Server (SSL Offload) dialog box, click SSL Settings, and then click SSL Policies.
12. In the Bind/Unbind SSL Policies dialog box, click Insert Policy. Under Policy Name, select the policy that you created in steps 5 through 7.
13. Click OK, and then click Close. A message appears in the status bar, stating that the policy has been bound successfully.
14. Repeat steps 12 and 13 and select the policy that you created in step 8.

Binding SSL Policies to a Virtual Server

Aug 20, 2013

The SSL policies that are configured on the NetScaler appliance need to be bound to a virtual server that intercepts traffic directed to the virtual server. If the incoming data matches any of the rules configured in the SSL policy, the policy is triggered and the action associated with it is carried out.

You can also bind SSL policies globally or to custom bind points on the NetScaler appliance. For more information about binding policies on the appliance, see Policy Configuration and Reference.

At the command prompt, type the following command to bind an SSL policy to a virtual server and verify the configuration:

- bind ssl vserver <vServerName> -policyName <string> [-priority <positive_integer>]
- show ssl vserver <vServerName>

Example

```
> bind ssl vserver vs-server -policyName ssl-policy-1 -priority 10
 Done
> show ssl vserver vs-server

        Advanced SSL configuration for VServer vs-server:
        DH: DISABLED
        Ephemeral RSA: ENABLED        Refresh Count: 1000
        Session Reuse: ENABLED        Timeout: 120 seconds
        Cipher Redirect: DISABLED
        SSLv2 Redirect: DISABLED
        ClearText Port: 80
        Client Auth: DISABLED
        SSL Redirect: ENABLED
        SSL-REDIRECT Port Rewrite: ENABLED
        Non FIPS Ciphers: DISABLED
        SSLv2: DISABLED SSLv3: ENABLED  TLSv1: ENABLED

1)     Policy Name: ssl-policy-1       Priority: 10

1)     Cipher Name: DEFAULT
       Description: Predefined Cipher Alias
 Done
```

1. Navigate to Traffic Management > SSL Offload > Virtual Servers.
2. In the details pane, from the list of virtual servers, select the virtual server to which you want to bind the SSL policy, and then click Open.
3. In the Configure Virtual Server (SSL Offload) dialog box, on the Policies tab, in the details pane, click Insert Policy.
4. Under Policy Name, select the policy that you want to bind to the virtual server.

5. Click OK, and then click Close. A message appears in the status bar, stating that the policy has been bound successfully.

Binding SSL Policies Globally

Aug 20, 2013

Globally bound policies are evaluated after all policies bound to services, virtual servers, or other NetScaler bind points are evaluated.

At the command prompt, type the following command to bind a global SSL policy and verify the configuration:

- bind ssl global - policyName <string> [- priority <positive_integer>]
- show ssl global

Example

```
> bind ssl global -policyName Policy-SSL-2 -priority 90
Done
> sh ssl global
1) Name: Policy-SSL-2 Priority: 90
2) Name: Policy-SSL-1 Priority: 100
Done
```

1. Navigate to Traffic Management > SSL > Policies.
2. In the details pane, click Global Bindings.
3. In the Bind/Unbind SSL Policies to Global dialog box, click Insert Policy.
4. In the Policy Name drop-down list, select a policy.
5. Optionally, drag the entry to a new position in the policy bank to automatically update the priority level.
6. Click OK. A message appears in the status bar, stating that the policy has been bound successfully.

Use Case 1: Configuring SSL Offloading with End-to-End Encryption

May 26, 2015

A simple SSL offloading setup terminates SSL traffic (HTTPS), decrypts the SSL records, and forwards the clear text (HTTP) traffic to the back-end web servers. However, the clear text traffic is vulnerable to being spoofed, read, stolen, or compromised by individuals who succeed in gaining access to the back-end network devices or web servers.

You can, therefore, configure SSL offloading with end-to-end security by re-encrypting the clear text data and using secure SSL sessions to communicate with the back-end Web servers.

Additionally, you can configure the back-end SSL transactions so that the NetScaler appliance uses SSL session multiplexing to reuse existing SSL sessions with the back-end web servers, thus avoiding CPU-intensive key exchange (full handshake) operations. This reduces the overall number of SSL sessions on the server, and therefore accelerates the SSL transaction while maintaining end-to-end security.

To configure SSL Offloading with end-to-end encryption, add SSL based services that represent secure servers with which the NetScaler appliance will carry out end-to-end encryption. Then create an SSL based virtual server, and create and bind a valid certificate-key pair to the virtual server. Bind the SSL services to the virtual server to complete the configuration.

For details on adding SSL based services, see Configuring Services.

For details on adding an SSL virtual server, see Configuring an SSL Based Virtual Server.

For details on creating a certificate-key pair, see Adding a Certificate-Key Pair.

For details on binding a certificate-key pair to a virtual server, see Binding the Certificate Key Pair to the SSL Based Virtual Server.

For details on binding services to a virtual server, see Binding Services to the SSL Based Virtual Server.

Example

Create two SSL based services, Service-SSL-1 and Service-SSL-2, with IP addresses 10.102.20.30 and 10.102.20.31 and both using port 443.

Then create an SSL based virtual server, Vserver-SSL-2 with an IP address of 10.102.10.20.

Next, create a certificate-key pair, CertKey-1 and bind it to the virtual server.

Bind the SSL services to the virtual server to complete the configuration.

Table 1. Entities in the SSL Offloading with End-to-End Encryption Example

Entity	Name	Value
SSL Service	Service-SSL-1	10.102.20.30
	Service-SSL-2	10.102.20.31

Entity SSL Based Virtual Server	Name Vserver-SSL-2	Value 10.102.10.20
Certificate - Key Pair	Certkey-1	

Use Case 2: Configuring Transparent SSL Acceleration

May 26, 2015

Note: You need to enable L2 mode on the NetScaler appliance for transparent SSL acceleration to work.
Transparent SSL acceleration is useful for running multiple applications on a secure server with the same public IP, and also for SSL acceleration without using an additional public IP.

In a transparent SSL acceleration setup, the NetScaler appliance is transparent to the client, because the IP address at which the appliance receives requests is the same as the Web server's IP address.

The NetScaler offloads SSL traffic processing from the Web server and sends either clear text or encrypted traffic (depending on the configuration) to the web server. All other traffic is transparent to the NetScaler and is bridged to the Web server. Therefore, other applications running on the server are unaffected.

There are three modes of transparent SSL acceleration available on the NetScaler:

- Service-based transparent access, where the service type can be SSL or SSL_TCP.
- Virtual server-based transparent access with a wildcard IP address (*:443).
- SSL VIP-based transparent access with end-to-end encryption.

Note: An SSL_TCP service is used for non-HTTPS services (for example SMTPS and IMAPS).

To enable transparent SSL acceleration using the SSL service mode, configure an SSL or an SSL_TCP service with the IP address of the actual back-end Web server. Instead of a virtual server intercepting SSL traffic and passing it on to the service, the traffic is now directly passed on to the service, which decrypts the SSL traffic and sends clear text data to the back-end server.

The service-based mode allows you to configure individual services with a different certificate, or with a different clear text port. Also, you can also select individual services for SSL acceleration.

You can apply service-based transparent SSL acceleration to data that uses different protocols, by setting the clear text port of the SSL service to the port on which the data transfer between the SSL service and the back-end server occurs.

To configure service-based transparent SSL acceleration, first enable both the SSL and the load balancing features. Then create an SSL based service and configure its clear text port. After the service is created, create and bind a certificate-key pair to this service.

For details on configuring the clear text port for an SSL based service, see "Configuring Advanced SSL Settings."

For details on creating a certificate-key pair and binding a certificate-key pair to a service, see "Adding a Certificate-Key Pair."

Example

Enable SSL offloading and load balancing.

Create an SSL based service, Service-SSL-1 with the IP address 10.102.20.30 using port 443 and configure its clear text port.

Next, create a certificate-key pair, CertKey-1 and bind it to the SSL service.

Table 1. Entities in the Service-based Transparent SSL Acceleration

Entity	Name	Value
SSL Service	Service-SSL-1	102.20.30
Certificate - Key Pair	Certkey-1	

You can use an SSL virtual server in the wildcard IP address mode if when you want to enable SSL acceleration for multiple servers that host the secure content of a Web site. In this mode, a single-digital certificate is enough for the entire secure Web site, instead of one certificate per virtual server. This results in significant cost savings on SSL certificates and renewals. The wildcard IP address mode also enables centralized certificate management.

To configure global transparent SSL acceleration on the NetScaler appliance, create a *:443 virtual server, which is a virtual server that accepts any IP address associated with port 443. Then, bind a valid certificate to this virtual server, and also bind all services to which the virtual server is to transfer. Such a virtual server can use the SSL protocol for HTTP-based data or the SSL_TCP protocol for non-HTTP-based data.

To configure virtual server-based acceleration with a wildcard IP address

1. Enable SSL, as described in "Enabling SSL Processing."
2. Enable load balancing, as described in "Load Balancing."
3. Add an SSL based virtual server (see "Configuring an SSL-Based Virtual Server" for the basic settings), and set the clearTextPort parameter (described in "Configuring Advanced SSL Settings)."
4. Add a certificate-key pair, as described in "Adding a Certificate-Key Pair."
   Note: The wildcard server will automatically learn the servers configured on the NetScaler, so you do not need to configure services for a wildcard virtual server.

Example

After enabling SSL offloading and load balancing, create an SSL based wildcard virtual server with IP address set to * and port number 443, and configure its clear text port (optional).

If you specify the clear text port, decrypted data will be sent to the backend server on that particular port. Otherwise, encrypted data will be sent to port 443.

Next, create an SSL certificate key pair, CertKey-1 and bind it to the SSL virtual server.

Table 2. Entities in the Virtual Server-based Acceleration with a Wildcard IP Address Example

Entity	Name	IP Address	Port
SSL Based Virtual Server	Vserver-SSL-Wildcard	*	443
Certificate - Key Pair	Certkey-1		

You can use an SSL virtual server for transparent access with end-to-end encryption if you have no clear text port specified. In such a configuration, the NetScaler terminates and offloads all SSL processing, initiates a secure SSL session, and sends the encrypted data, instead of clear text data, to the web servers on the port that is configured on the wildcard virtual server.

Note: In this case, the SSL acceleration feature runs at the back-end, using the default configuration, with all 34 ciphers available.
To configure SSL VIP based transparent access with end-to-end encryption, Follow instructions for Configuring a Virtual Server-based Acceleration with a Wildcard IP Address (*:443), but do not configure a clear text port on the virtual server.

Use Case 3: Configuring SSL Acceleration with HTTP on the Front End and SSL on the Back End

May 26, 2015

In certain deployments, you might be concerned about network vulnerabilities between the NetScaler appliance and the backend servers, or you might need complete end-to-end security and interaction with certain devices that can communicate only in clear text (for example, caching devices).

In such cases, you can set up an HTTP virtual server that receives data from clients that connect to it at the front end and hands the data off to a secure service, which securely transfers the data to the web server.

To implement this type of configuration, you configure an HTTP virtual server on the NetScaler and bind SSL based services to the virtual server. The NetScaler receives HTTP requests from the client on the configured HTTP virtual server, encrypts the data, and sends the encrypted data to the web servers in a secure SSL session.

To configure SSL acceleration with HTTP on the front-end and SSL on the back-end, first enable the load balancing and SSL features on the NetScaler. Then, add SSL based services that represent secure servers to which the NetScaler appliance will send encrypted data. Finally, add an HTTP based virtual server and bind the SSL services to this virtual server.

Example

Enable load balancing and SSL acceleration on the NetScaler.

After enabling load balancing and SSL acceleration, create two SSL based services, Service-SSL-1 and Service-SSL-2, with IP addresses 10.102.20.30 and 10.102.20.31, and both using port 443.

Then create an HTTP based virtual server, Vserver-HTTP-1, with an IP address of 10.102.10.20.

Bind the SSL services to the virtual server to complete the configuration.

Table 1. Entities in the SSL Acceleration with HTTP on the Front End and SSL on the Back End Example

Entity	Name	Value
SSL Service	Service-SSL-1	10.102.20.30
	Service-SSL-2	10.102.20.31
HTTP Based Virtual Server	Vserver-HTTP-1	10.102.10.20

Use Case 4: SSL Offloading with Other TCP Protocols

May 26, 2015

In addition to the secure HTTP (HTTPS) protocol, NetScaler appliances support SSL acceleration for other TCP-based secure protocols. However, only simple requests and response-based TCP application protocols are supported. Applications such as FTPS, that insert the server's IP address and port information in their payloads, are not currently supported.

Note: The STARTTLS feature for SMTP is currently not supported.
The NetScaler supports SSL acceleration for Other TCP protocols with and without end-to-end encryption.

To configure SSL offloading with Other TCP protocols, create a virtual server of type SSL_TCP, bind a certificate-key pair and TCP based services to the virtual server, and configure SSL actions and policies based on the type of traffic expected and the acceleration to be provided.

Follow the instructions in Configuring SSL Offloading, but create an SSL_TCP virtual server instead of an SSL virtual server, and configure TCP services instead of HTTP services.

To configure SSL_TCP-based offloading with end-to-end encryption, both the virtual server that intercepts secure traffic and the services that it forwards the traffic to must be of type SSL_TCP.

Configure SSL_TCP-based offloading as described in Configuring SSL Offloading with End-to-End Encryption, but create an SSL_TCP virtual server instead of an SSL virtual server.

Some deployments might require the NetScaler appliance to encrypt TCP data received as clear text and send the data securely to the back end servers.

To provide SSL acceleration with back-end encryption for clear text TCP traffic arriving from the client, create a TCP based virtual server and bind it to SSL_TCP based services.

To configure end-to-end encryption for TCP-based data, follow the procedure described in Configuring the SSL feature with HTTP on the Front-End and SSL on the Back-End, but create a TCP virtual server instead of an HTTP virtual server.

Use Case 5: Configuring SSL Bridging

May 26, 2015

An SSL bridge configured on the NetScaler appliance enables the appliance to bridge all secure traffic between the SSL client and the SSL server. The appliance does not offload or accelerate the bridged traffic, nor does it perform encryption or decryption. Only load balancing is done by the appliance. The SSL server must handle all SSL-related processing. Features such as content switching, SureConnect, and cache redirection do not work, because the traffic passing through the appliance is encrypted.

Because the appliance does not carry out any SSL processing in an SSL bridging setup, there is no need for SSL certificates.

Citrix recommends that you use this configuration only if an acceleration unit (for example, a PCI-based SSL accelerator card) is installed in the web server to handle the SSL processing overhead.

Before you configure SSL bridging, first enable SSL and load balancing on the appliance. Then, create SSL_Bridge services and bind them to an SSL_Bridge virtual server. Configure the load balancing feature to maintain server persistency for secure requests.

Example

After enabling SSL and load balancing, create two servers, s1 and s2. Create two SSL_Bridge services, sc1 and src2. Create an SSL_Bridge virtual server and bind the SSL_Bridge services to the virtual server to complete the configuration. At the command line, type:

```
enable ns feature SSL LB
add server s1 10.102.1.101
add server s2 10.102.1.102
add service src1 s1 SSL_BRIDGE 443
add service src2 s2 SSL_BRIDGE 443
add lb vserver ssl_bridge_vip SSL_BRIDGE 10.102.1.200 443
bind lb vserver ssl_bridge_vip src1
bind lb vserver ssl_bridge_vip src2
```

Use Case 6: Configuring SSL Monitoring when Client Authentication is Enabled on the Backend Service

May 26, 2015

Consider a scenario in which you need to load balance servers that require SSL client certificates to validate clients. For this deployment, you need to create an SSL service on the NetScaler appliance, add an HTTPS monitor, add a certificate-key pair, bind this certificate-key pair to the SSL service, and then bind the https monitor to this service. You can use this https monitor to perform health checks on the backend services.

1. Open an SSH connection to the appliance by using an SSH client, such as PuTTY.
2. Log on the appliance by using the administrator credentials.
3. Add an SSL service. At the command prompt, type:
   add service <name> <serverName> <serviceType> <port>
4. Add an https monitor. At the command prompt, type:
   add lb monitor <name> <type>
5. Add the certificate-key pair that is going to be used as the client cert for that SSL service. At the command prompt, type:
   add ssl certKey <certkeyName> -cert <string> -key <string>
6. Bind this certkey to the SSL service. At the command prompt, type:
   bind ssl service <serviceName> -certkeyName <string>
7. Bind the https monitor to the SSL service. At the command prompt, type:
   bind lb monitor <monitorName> <serviceName>

Now, when the appliance tries to probe the backend service on which client authentication is enabled, the backend service will request a certificate as part of the SSL handshake. When the appliance returns the certificate-key bound in step 6 above, the monitor probe will succeed.

Example

```
add service svc_k 10.102.145.30 SSL 443
add lb monitor sslmon HTTP -respCode 200 -httpRequest "GET /testsite/file5.html" -secure YES
add ssl certKey ctest -cert client_rsa_2048.pem -key client_rsa_2048.ky
bind ssl service svc_k -certkeyName ctest
bind lb monitor sslmon svc_k
> show service svc_k
    svc_k (10.102.145.30:443) - SSL
    State: UP
    Last state change was at Tue Jan 10 13:12:24 2012
    Time since last state change: 0 days, 00:09:37.890
    Server Name: 10.102.145.30
    Server ID : 0 Monitor Threshold : 0
    Max Conn: 0 Max Req: 0 Max Bandwidth: 0 kbits
```

```
    Use Source IP: NO
    Client Keepalive(CKA): NO
    Access Down Service: NO
    TCP Buffering(TCPB): NO
    HTTP Compression(CMP): NO
    Idle timeout: Client: 180 sec Server: 360 sec
    Client IP: DISABLED
    Cacheable: NO
    SC: OFF
    SP: OFF
    Down state flush: ENABLED
    Appflow logging: ENABLED

1) Monitor Name: sslmon
    State: UP Weight: 1
    Probes: 1318 Failed [Total: 738 Current: 0]
    Last response: Success - HTTP response code 200 received.
    Response Time: 0.799 millisec
  Done
>
> show ssl service svc_k
    Advanced SSL configuration for Back-end SSL Service svc_k:
    DH: DISABLED
    Ephemeral RSA: DISABLED
    Session Reuse: ENABLED Timeout: 300 seconds
    Cipher Redirect: DISABLED
    SSLv2 Redirect: DISABLED
    Server Auth: DISABLED
    SSL Redirect: DISABLED
    Non FIPS Ciphers: DISABLED
    SNI: DISABLED
    SSLv2: DISABLED SSLv3: ENABLED TLSv1: ENABLED
1) CertKey Name: ctest Client Certificate

1) Cipher Name: ALL
  Description: Predefined Cipher Alias
Done
```

Use Case 7: Configuring a Secure Content Switching Server

May 26, 2015

An SSL-based content switching virtual server first decrypts the secure data and then redirects the data to appropriately configured servers as determined by the type of content and the configured content switching policies. The packets sent to the server have a mapped IP address as the source IP address.

The following example shows the steps to configure two address-based virtual servers to perform load balancing on the HTTP services. One virtual server, Vserver-LB-HTML, load balances the dynamic content (cgi, asp), and the other, Vserver-LB-Image, load balances the static content (gif, jpeg). The load-balancing method used is the default, LEASTCONNECTION. A content switching SSL virtual server, Vserver-CS-SSL, is then configured to perform SSL acceleration and switching of HTTPS requests on the basis of configured content switching policies.

Example

```
> enable ns feature lb cs ssl
> add lb vserver Vserver-LB-HTML http 10.1.1.2 80
> add lb vserver Vserver-LB-Image http 10.1.1.3 80
> add service s1 10.1.1.4 http 80
> add service s2 10.1.1.5 http 80
> add service s3 10.1.1.6 http 80
> add service s4 10.1.1.7 http 80
> bind lb vserver Vserver-LB-HTML s1
> bind lb vserver Vserver-LB-HTML s2
> bind lb vserver Vserver-LB-Image s3
> bind lb vserver Vserver-LB-Image s4
> add cs vserver Vserver-CS-SSL ssl 10.1.1.1 443
> add cs policy pol1 -url "*.cgi"
> add cs policy pol2 -url "*.asp"
> add cs policy pol3 -url "*.gif"
> add cs policy pol4 -url "*.jpeg"
> bind cs vserver Vserver-CS-SSL -policyName pol1 Vserver-LB-HTML
> bind cs vserver Vserver-CS-SSL -policyName pol2 Vserver-LB-HTML
> bind cs vserver Vserver-CS-SSL -policyName pol3 Vserver-LB-Image
> bind cs vserver Vserver-CS-SSL -policyName pol4 Vserver-LB-Image
> add certkey mykey -cert /nsconfig/ssl/ns-root.cert -key /nsconfig/ssl/ns-root.key
> bind certkey Vserver-CS-SSL mykey
>
> show cs vserver Vserver-CS-SSL
      Vserver-CS-SSL (10.1.1.1:443) - SSL Type: CONTENT
      State: UP
      Last state change was at Tue Jul 13 02:11:37 2010
      Time since last state change: 0 days, 00:02:12.440
      Client Idle Timeout: 180 sec
      Down state flush: ENABLED
```

```
Disable Primary Vserver On Down : DISABLED
State Update: DISABLED
Default:        Content Precedence: RULE
Vserver IP and Port insertion: OFF
Case Sensitivity: ON
Push: DISABLED  Push VServer:
Push Label Rule: none
```

Ciphers Supported by the NetScaler Appliance

May 13, 2015

Your NetScaler appliance ships with a predefined set of cipher groups. Table 1 lists the ciphers that are part of the DEFAULT cipher group and are therefore bound by default to an SSL virtual server. Table 2 lists the other ciphers currently supported by the NetScaler appliance. To use ciphers that are not part of the DEFAULT cipher group, you have to explicitly bind them to an SSL virtual server. You can also create a user-defined cipher group to bind to the SSL virtual server. For more information about creating a user-defined cipher group, see Configuring User-Defined Cipher Groups on the NetScaler Appliance.

Note:

- On all NetScaler appliances, connection between a NetScaler service and the back end server can be established using TLS protocol version 1.0 or SSLv3 protocol only.
- Support for TLS versions 1.1 and 1.2 is only available on the front end (between client and virtual server) on the following appliances:
  - NetScaler MPX appliances.
  - NetScaler SDX appliances on an instance-by-instance basis. To support TLS protocol versions 1.1 and 1.2 on an SDX appliance, you must assign at least one SSL chip to the instance when you provision it.
- Support for TLS protocol versions 1.1 and 1.2 is not available on a FIPS appliance or on a NetScaler virtual appliance.

Table 1. Ciphers That the NetScaler Appliance Supports by Default

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm	Hex Code
SSL3-RC4-MD5	SSLv3 TLSv1 TLSv1.1 TLSv1.2	RSA	RSA	RC4(128)	MD5	
SSL3-RC4-SHA	SSLv3 TLSv1 TLSv1.1 TLSv1.2	RSA	RSA	RC4(128)	SHA1	
SSL3-DES-CBC3-SHA	SSLv3 TLSv1 TLSv1.1	RSA	RSA	3DES(168)	SHA1	

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm	Hex Code
TLS1-AES-256-CBC-SHA	SSLv3 TLSv1 TLSv1.1 TLSv1.2	RSA	RSA	AES(256)	SHA1	
TLS1-AES-128-CBC-SHA	SSLv3 TLSv1 TLSv1.1 TLSv1.2	RSA	RSA	AES(128)	SHA1	
SSL3-EDH-DSS-DES-CBC3-SHA	SSLv3 TLSv1	DH	DSS	3DES(168)	SHA1	
TLS1-DHE-DSS-RC4-SHA	TLSv1	DH	DSS	RC4(128)	SHA1	
TLS1-DHE-DSS-AES-256-CBC-SHA	SSLv3 TLSv1	DH	DSS	AES(256)	SHA1	
TLS1-DHE-DSS-AES-128-CBC-SHA	SSLv3 TLSv1	DH	DSS	AES(128)	SHA1	
SSL3-EDH-RSA-DES-CBC3-SHA	SSLv3 TLSv1 TLSv1.1 TLSv1.2	DH	RSA	3DES(168)	SHA1	
TLS1-DHE-RSA-AES-256-CBC-SHA	SSLv3 TLSv1 TLSv1.1	DH	RSA	AES(256)	SHA1	

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm	Hex Code
TLS1-DHE-RSA-AES-128-CBC-SHA	TLSv1.2 SSLv3 TLSv1 TLSv1.1 TLSv1.2	DH	RSA	AES(128)	SHA1	

Table 2. Additional Ciphers Supported by the NetScaler Appliance

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm
SSL3-DES-CBC-SHA	SSLv3 TLSv1 TLSv1.1	RSA	RSA	DES(56)	SHA1
TLS1-EXP1024-RC4-SHA	TLSv1	RSA(1024)	RSA	RC4(56)	SHA1 Export
SSL3-EXP-RC4-MD5	SSLv3 TLSv1	RSA(512)	RSA	RC4(40)	MD5 Export
SSL3-EXP-DES-CBC-SHA	SSLv3 TLSv1	RSA(512)	RSA	DES(40)	SHA1 Export
SSL3-EXP-RC2-CBC-MD5	SSLv3 TLSv1	RSA(512)	RSA	RC2(40)	MD5 Export
SSL2-RC4-MD5	SSLv2	RSA	RSA	RC4(128)	MD5
SSL2-DES-CBC3-MD5	SSLv2	RSA	RSA	3DES(168)	MD5
SSL2-RC2-CBC-MD5	SSLv2	RSA	RSA	RC2(128)	MD5
SSL2-DES-CBC-MD5	SSLv2	RSA	RSA	DES(56)	MD5

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm
SSL2-RC4-64-MD5	SSLv2	RSA	RSA	RC4(64)	MD5
SSL2-EXP-RC4-MD5	SSLv2	RSA(512)	RSA	RC4(40)	MD5 Export
SSL3-EDH-DSS-DES-CBC-SHA	SSLv3 TLSv1	DH	DSS	DES(56)	SHA1
TLS1-EXP1024-DHE-DSS-DES-CBC- SHA	TLSv1	DH(1024)	DSS	DES(56)	SHA1 Export
TLS1-EXP1024-DHE-DSS-RC4- SHA	TLSv1	DH(1024)	DSS	RC4(56)	SHA1 Export
SSL3-EXP-EDH-DSS-DES-CBC-SHA	SSLv3 TLSv1	DH(512)	DSS	DES(40)	SHA1 Export
SSL3-EDH-RSA-DES-CBC-SHA	SSLv3 TLSv1 TLSv1.1	DH	RSA	DES(56)	SHA1
SSL3-EXP-EDH-RSA-DES-CBC-SHA	SSLv3 TLSv1	DH(512)	RSA	DES(40)	DES(40)
TLS1-EXP1024-RC4-MD5	TLSv1	RSA(1024)	RSA	RC4(56)	MD5 Export
TLS1-EXP1024-RC2-CBC-MD5	TLSv1	RSA(1024)	RSA	RC2(56)	MD5 Export
SSL2-EXP-RC2-CBC-MD5	SSLv2	RSA(512)	RSA	RC2(40)	MD5 Export
SSL3-ADH-RC4-MD5	SSLv3 TLSv1 TLSv1.1	DH	None	RC4(128)	MD5

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm
SSL3-ADH-DES-CBC-SHA	SSLv3 TLSv1 TLSv1.1	DH	None	DES(56)	SHA1
SSL3-ADH-DES-CBC3-SHA	SSLv3 TLSv1 TLSv1.1	DH	None	3DES(168)	SHA1
TLS1-ADH-AES-128-CBC-SHA	SSLv3 TLSv1 TLSv1.1	DH	None	AES(128)	SHA1
TLS1-ADH-AES-256-CBC-SHA	SSLv3 TLSv1 TLSv1.1	DH	None	AES(256)	SHA1
SSL3-EXP-ADH-RC4-MD5	SSLv3 TLSv1	DH(512)	None	RC4(40)	MD5 Export
SSL3-EXP-ADH-DES-CBC-SHA	SSLv3 TLSv1	DH(512)	None	DES(40)	SHA1 Export
TLS1-ECDHE-RSA-RC4-SHA	TLSv1 TLSv1.1 TLSv1.2	ECC-DHE	RSA	RC4(128)	SHA1
TLS1-ECDHE-RSA-DES-CBC3-SHA	TLSv1 TLSv1.1 TLSv1.2	ECC-DHE	RSA	3DES(168)	SHA1
TLS1-ECDHE-RSA-AES128-SHA	TLSv1 TLSv1.1 TLSv1.2	ECC-DHE	RSA	AES(128)	SHA1

Cipher Suite	Protocol	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm (Key Size)	Message Authentication Code (MAC) Algorithm
TLS1-ECDHE-RSA-AES256-SHA	TLSv1 TLSv1.1	ECC-DH2	RSA	AES(256)	SHA1
	TLSv1.2				

On a NetScaler platform that does not have N3 chips and is configured to negotiate EDH ciphers by using TLS version 1.0 with a DH key of 2048 bits, the SSL handshake fails in either of the following scenarios:

- Client authentication is enabled and the appliance receives a client certificate of 4096 bits.
- End-to-end encryption is configured and the appliance receives a server certificate of 4096 bits.

Use the show ns hardware command to find out if your appliance has N3 chips.

Example

```
> sh hardware
        Platform: NSMPX-22000 16*CPU+24*IX+12*E1K+2*E1K+4*CVM N3 2200100
        Manufactured on: 8/19/2013
        CPU: 2900MHZ
        Host Id: 1006665862
        Serial no: ENUK6298FT
        Encoded serial no: ENUK6298FT
 Done
```

FIPS

Jun 09, 2015

The Federal Information Processing Standard (FIPS), issued by the US National Institute of Standards and Technologies, specifies the security requirements for a cryptographic module used in a security system. The NetScaler FIPS appliance complies with the second version of this standard, FIPS-140-2.

Note: Henceforth, all references to FIPS imply FIPS-140-2.
The FIPS appliance is equipped with a tamper-proof (tamper-evident) cryptographic module—and a Cavium CN1620-NFBE3-2.0-G on the MPX 9700/10500/12500/15500 FIPS appliances—designed to comply with the FIPS 140-2 Level-2 specifications. The Critical Security Parameters (CSPs), primarily the server's private-key, are securely stored and generated inside the cryptographic module, also referred to as the Hardware Security Module (HSM). The CSPs are never accessed outside the boundaries of the HSM. Only the superuser (nsroot) can perform operations on the keys stored inside the HSM.

The following table summarizes the differences between standard NetScaler and NetScaler FIPS appliances.

Setting	NetScaler appliance	NetScaler FIPS appliance
Key storage	On the hard disk	On the FIPS card
Cipher support	All ciphers	FIPS approved ciphers
Accessing keys	From the hard disk	Not accessible

Configuring a FIPS appliance involves configuring the HSM immediately after completing the generic configuration process. You then create or import a FIPS key. After creating a FIPS key, you should export it for backup. You might also need to export a FIPS key so that you can import it to another appliance. For example, configuring FIPS appliances in a high availability (HA) setup requires transferring the FIPS key from the primary node to the secondary node immediately after completing the standard HA setup.

You can upgrade the firmware version on the FIPS card from version 4.6.0 to 4.6.1, and you can reset an HSM that has been locked to prevent unauthorized logon. Only FIPS approved ciphers are supported on a NetScaler FIPS appliance.

This section includes the following details:

- Configuring the HSM
- Creating and Transferring FIPS Keys
- Configuring FIPS Appliances in a High Availability Setup
- Resetting a Locked HSM
- FIPS Approved Algorithms and Ciphers

Configuring the HSM

May 07, 2015

Before you can configure the HSM of your NetScaler FIPS appliance, you must complete the initial hardware configuration. For more information, see Initial Configuration.

Configuring the HSM of your NetScaler FIPS appliance erases all existing data on the HSM. To configure the HSM, you must be logged on to the appliance as the superuser (nsroot account). The HSM is preconfigured with default values for the Security Officer (SO) password and User password, which you use to configure the HSM or reset a locked HSM. The maximum length allowed for the password is 14 alphanumeric characters. Symbols are not allowed.
Important: Do not perform the set ssl fips command without first resetting the FIPS card and restarting the MPX FIPS appliance.
Although the FIPS appliance can be used with the default password values, you should modify them before using it. The HSM can be configured only when you log on to the appliance as the superuser and specify the SO and User passwords.

Important: Due to security constraints, the appliance does not provide a means for retrieving the SO password. Store a copy of the password safely. Should you need to reinitialize the HSM, you will need to specify this password as the old SO password.
Before initializing the HSM, you can upgrade to the latest build of the software. To upgrade to the latest build, see Upgrading or Downgrading the System Software.

After upgrading, verify that the /nsconfig/fips directory has been successfully created on the appliance.

After logging on to the appliance as the superuser and completing the initial configuration, at the command prompt, type the following commands to configure the HSM and verify the configuration:

1. show ssl fips
2. reset ssl fips
3. reboot
4. set ssl fips -initHSM Level-2 <newSOpassword> <oldSOpassword> <userPassword> [-hsmLabel <string>]
5. save ns config
6. reboot
7. show ssl fips

Example

```
show fips
FIPS Card is not configured
Done
reset fips
reboot
Are you sure you want to restart NetScaler (Y/N)? [N]:y
set ssl fips -initHSM Level-2 sopin12345 so12345 user123 -hsmLabel cavium
This command will erase all data on the FIPS card. You must save the configuration
(saveconfig) after executing this command.
```

```
Do you want to continue?(Y/N)y
Done
save ns config
reboot
Are you sure you want to restart NetScaler (Y/N)? [N]:y
show fips
      FIPS HSM Info:
HSM Label           : NetScaler FIPS
Initialization      : FIPS-140-2 Level-2
HSM Serial Number   : 2.1G1008-IC000021
HSM State           : 2

Firmware Version    : 1.1
Firmware Release Date  : Jun04,2010

Max FIPS Key Memory   : 3996
Free FIPS Key Memory  : 3994
Total SRAM Memory     : 467348
Free SRAM Memory      : 62564
Total Crypto Cores    : 3
Enabled Crypto Cores  : 1
Done
```

Note: In Release 10.5.e build xx.x and later, the **Firmware Release Date** in the above output is replaced with the **Firmware Build**. For example, Firmware Build : NFBE-FW-2.2-130009.

1. Navigate to Traffic Management > SSL > FIPS.
2. In the details pane, on the FIPS Infotab, click Reset FIPS.
3. In the navigation pane, click System.
4. In the details pane, click Reboot.
5. In the details pane, on the FIPS Info tab, click Initialize HSM.
6. In the Initialize HSM dialog box, specify values for the following parameters:
   - Security Officer (SO) Password*—new SO password
   - Old SO Password*—old SO password
   - User Password*—user password
   - Level—initHSM (Currently set to Level2 and cannot be changed)
   - HSM Label—hsmLabel

   *A required parameter
7. Click OK.
8. In the details pane, click Save.
9. In the navigation pane, click System.
10. In the details pane, click Reboot.
11. Under FIPS HSM Info, verify that the information displayed for the FIPS HSM that you just configured is correct.

Creating and Transferring FIPS Keys

Dec 22, 2014

After configuring the HSM of your FIPS appliance, you are ready to create a FIPS key. The FIPS key is created in the appliance's HSM. You can then export the FIPS key to the appliance's CompactFlash card as a secured backup. Exporting the key also enables you to transfer it by copying it to the /flash of another appliance and then importing it into the HSM of that appliance.

Instead of creating a FIPS key, you can import an existing FIPS key or import an external key as a FIPS key. If you are adding a certificate-key pair of 2048 bits on the MPX 9700/10500/12500/15500 FIPS appliances, make sure that you have the correct certificate and key pair.

Note: If you are planning an HA setup, make sure that the FIPS appliances are configured in an HA setup before creating a FIPS key.

Updated: 2013-08-20

Before creating a FIPS key, make sure that the HSM is configured.

To create a FIPS key by using the configuration utility

1. Navigate to Traffic Management > SSL > FIPS.
2. In the details pane, on the FIPS Keys tab, click Add.
3. In the Create FIPS Key dialog box, specify values for the following parameters:
   - FIPS Key Name*—fipsKeyName
   - Modulus*—modulus
   - Exponent*—exponent

   *A required parameter
4. Click Create, and then click Close.
5. On the FIPS Keys tab, verify that the settings displayed for the FIPS key that you just created are correct.

To create a FIPS key by using the command line interface

At the command prompt, type the following commands to create a FIPS key and verify the settings:

- create ssl fipsKey <fipsKeyName> -modulus <positive_integer> [-exponent (3 | F4)]
- show ssl fipsKey [<fipsKeyName>]

Example

```
create fipskey Key-FIPS-1 -modulus 2048 -exponent 3
show ssl fipsKey Key-FIPS-1
FIPS Key Name: Key-FIPS-1 Modulus: 2048  Public Exponent: 3 (Hex: 0x3)
```

Updated: 2013-08-20

Citrix recommends that you create a backup of any key created in the FIPS HSM. If a key in the HSM is deleted, there is no

way to create the same key again, and all the certificates associated with it are rendered useless.

In addition to exporting a key as a backup, you might need to export a key for transfer to another appliance.

The following procedure provides instructions on exporting a FIPS key to the /nsconfig/ssl folder on the appliance's CompactFlash and securing the exported key by using a strong asymmetric key encryption method.

To export a FIPS key by using the command line interface

At the command prompt, type:
export ssl fipsKey <fipsKeyName> -key <string>
Example

```
export fipskey Key-FIPS-1 -key Key-FIPS-1.key
```

To export a FIPS key by using the configuration utility

1. Navigate to Traffic Management > SSL > FIPS
2. In the details pane, on the FIPS Keys tab, click Export.
3. In the Export FIPS key to a file dialog box, specify values for the following parameters:
   - FIPS Key Name*—fipsKeyName
   - File Name*—key (To put the file in a location other than the default, you can either specify the complete path or click the Browse button and navigate to a location.)

   *A required parameter
4. Click Export, and then click Close.

Updated: 2013-11-22

To use an existing FIPS key with your FIPS appliance, you need to transfer the FIPS key from the hard disk of the appliance into its HSM.

Note: To avoid errors when importing a FIPS key, make sure that the name of the key imported is the same as the original key name when it was created.

To import a FIPS key on the MPX 9700/10500/12500/15500 FIPS appliances by using the command line interface

At the command prompt, type the following commands to import a FIPS key and verify the settings:
- import ssl fipsKey <fipsKeyName> -key <string> -inform SIM -exponent (F4 | 3)
- show ssl fipskey <fipsKeyName>

Example

```
import fipskey Key-FIPS-2 -key Key-FIPS-2.key -inform SIM -exponent F4
show ssl fipskey key-FIPS-2
FIPS Key Name: Key-FIPS-2 Modulus: 2048  Public Exponent: F4 (Hex value 0x10001)
```

To import a FIPS key by using the configuration utility

1. Navigate to Traffic Management > SSL > FIPS
2. In the details pane, on the FIPS Keys tab, click Import.

3. In the Import as a FIPS Key dialog box, select FIPS key file and set values for the following parameters:
   - FIPS Key Name*
   - Key File Name*—To put the file in a location other than the default, you can either specify the complete path or click Browse and navigate to a location.
   - Exponent*

   *A required parameter
4. Click Import, and then click Close.
5. On the FIPS Keys tab, verify that the settings displayed for the FIPS key that you just imported are correct.

Updated: 2015-02-09

In addition to transferring FIPS keys that are created within the NetScaler appliance's HSM, you can transfer external private keys (such as those created on a standard NetScaler, Apache, or IIS) to a FIPS NetScaler appliance. External keys are created outside the HSM, by using a tool such as OpenSSL. Before importing an external key into the HSM, copy it to the appliance's flash drive under /nsconfig/ssl.

Importing an external key as a FIPS key on the MPX 9700/10500/12500/15500 FIPS appliances by using the command line interface

On the MPX 9700/10500/12500/15500 FIPS appliances, the -exponent parameter in the import ssl fipskey command is not required while importing an external key. The correct public exponent is detected automatically when the key is imported, and the value of the -exponent parameter is ignored.

The NetScaler FIPS appliance does not support external keys with a public exponent other than 3 or F4.

You do not need a wrap key on the MPX 9700/10500/12500/15500 FIPS appliances.

You cannot import an external, encrypted FIPS key directly to an MPX 9700/10500/12500/15500 FIPS appliance. To import the key you need to first decrypt the key, and then import it. To decrypt the key, at the shell prompt, type:

openssl rsa -in <EncryptedKey.key> > <DecryptedKey.out>

1. Copy the external key to the appliance's flash drive.
2. If the key is in .pfx format, you must first convert it to PEM format. At the command prompt, type:
   - convert ssl pkcs12 <output file> -import -pkcs12File <input .pfx file name> -password <password>
3. At the command prompt, type the following commands to import the external key as a FIPS key and verify the settings:
   - import ssl fipsKey <fipsKeyName> -key <string> -inform PEM
   - show ssl fipskey<fipsKeyName>

Example

```
convert ssl pkcs12 iis.pem -password 123456 -import -pkcs12File iis.pfx
import fipskey Key-FIPS-2 -key iis.pem -inform PEM
show ssl fipskey key-FIPS-2
FIPS Key Name: Key-FIPS-2 Modulus: 0   Public Exponent: F4 (Hex value 0x10001)
```

Note: The modulus is incorrectly displayed as zero in the above example. The discrepancy does not affect SSL functionality.

To import an external key as a FIPS key to an MPX 9700/10500/12500/15500 FIPS appliance by using the configuration utility

1. If the key is in .pfx format, you must first convert it to PEM format.
   1. Navigate to Traffic Management > SSL.
   2. In the details pane, under Tools, click Import PKCS#12.
   3. In the Import PKCS12 File dialog box, set the following parameters:
      - Output File Name*
      - PKCS12 File Name*—Specify the .pfx file name.
      - Import Password*
      - Encoding Format

      *A required parameter
2. Navigate to Traffic Management > SSL > FIPS
3. In the details pane, on the FIPS Keys tab, click Import.
4. In the Import as a FIPS Key dialog box, select PEM file, and set values for the following parameters:
   - FIPS Key Name*
   - Key File Name*—To put the file in a location other than the default, you can either specify the complete path or click Browse and navigate to a location.

   *A required parameter
5. Click Import, and then click Close.
6. On the FIPS Keys tab, verify that the settings displayed for the FIPS key that you just imported are correct.

Configuring FIPS Appliances in a High Availability Setup

Nov 11, 2013

You can configure two appliances in a high availability (HA) pair as FIPS appliances. For information about configuring an HA setup, see High Availability.

Note: Citrix recommends that you use the configuration utility (GUI) for this procedure. If you use the command line (CLI), make sure that you carefully follow the steps as listed in the procedure. Changing the order of steps or specifying an incorrect input file might cause inconsistency that requires that the appliance be restarted. In addition, if you use the CLI, the create ssl fipskey command is not propagated to the secondary node. When you execute the command with the same input values for modulus size and exponent on two different FIPS appliances, the keys generated are not identical. You have to create the FIPS key on one of the nodes and then transfer it to the other node. But if you use the configuration utility to configure FIPS appliances in an HA setup, the FIPS key that you create is automatically transferred to the secondary node. The process of managing and transferring the FIPS keys is known as secure information management (SIM).
Important: On the MPX 9700/10500/12500/15500 FIPS appliances, the HA setup should be completed within six minutes. If the process takes longer than six minutes, the internal timer of the FIPS card expires and the following error message appears:
ERROR: Operation timed out or repeated, please wait for 10 mins and redo the SIM/HA configuration steps.

If this message appears, restart the appliance or wait for 10 minutes, and then repeat the HA setup procedure.

In the following procedure, appliance A is the primary node and appliance B is the secondary node.

1. **On appliance A**, open an SSH connection to the appliance by using an SSH client, such as PuTTY.
2. Log on to the appliance, using the administrator credentials.
3. Initialize appliance A as the source appliance. At the command prompt, type:
   init ssl fipsSIMsource <certFile>
4. Copy this <certFile> file to appliance B, in the /nconfig/ssl folder.
5. **On appliance B**, open an SSH connection to the appliance by using an SSH client, such as PuTTY.
6. Log on to the appliance, using the administrator credentials.
7. Initialize appliance B as the target appliance. At the command prompt, type:
   init ssl fipsSIMtarget <certFile> <keyVector> <targetSecret>
8. Copy this <targetSecret> file to appliance A.
9. **On appliance A**, enable appliance A as the source appliance. At the command prompt, type:
   enable ssl fipsSIMSource <targetSecret> <sourceSecret>
10. Copy this <sourceSecret> file to appliance B.
11. **On appliance B**, enable appliance B as the target appliance. At the command prompt, type:
    enable ssl fipsSIMtarget <keyVector> <sourceSecret>
12. **On appliance A**, create a FIPS key, as described in Creating a FIPS Key.
13. Export the FIPS key to the appliance's hard disk, as described in Exporting a FIPS Key.

14. Copy the FIPS key to the hard disk of the secondary appliance by using a secure file transfer utility, such as SCP.
15. **On appliance B**, import the FIPS key from the hard disk into the HSM of the appliance, as described in Importing an Existing FIPS Key.

1. On the appliance to be configured as the source appliance, navigate to Traffic Management > SSL > FIPS.
2. In the details pane, on the FIPS Info tab, click Enable SIM.
3. In the Enable HA Pair for SIM dialog box, in the Certificate File Name text box, type the file name, with the path to the location at which the FIPS certificate should be stored on the source appliance.
4. In the Key Vector File Name text box, type the file name, with the path to the location at which the FIPS key vector should be stored on the source appliance.
5. In the Target Secret File Name text box, type the location for storing the secret data on the target appliance.
6. In the Source Secret File Name text box, type the location for storing the secret data on the source appliance.
7. Click OK. The FIPS appliances are now configured in HA mode.
8. Create a FIPS key, as described in Creating a FIPS Key. The FIPS key is automatically transferred from the primary to the secondary.

Example

In the following example, source.cert is the certificate on the source appliance, stored in the default directory, /nsconfig/ssl. This certificate must be transferred to the same location (/nsconfig/ssl) on the target appliance. The file target.secret is created on the target appliance and copied to the source appliance. The file source.secret is created on the source appliance and copied to the target appliance.

On the source appliance

```
init fipsSIMsource /nsconfig/ssl/source.cert
```

On the target appliance

```
init fipsSIMtarget /nsconfig/ssl/source.cert /nsconfig/ssl/target.key /nsconfig/ssl/target.secret
```

On the source appliance

```
enable fipsSIMsource /nsconfig/ssl/target.secret /nsconfig/ssl/source.secret
```

On the target appliance

```
enable fipsSIMtarget /nsconfig/ssl/target.key /nsconfig/ssl/source.secret
```

On the source appliance

```
create ssl fipskey fips1 -modulus 2048 -exponent f4
export fipskey fips1 -key /nsconfig/ssl/fips1.key
```

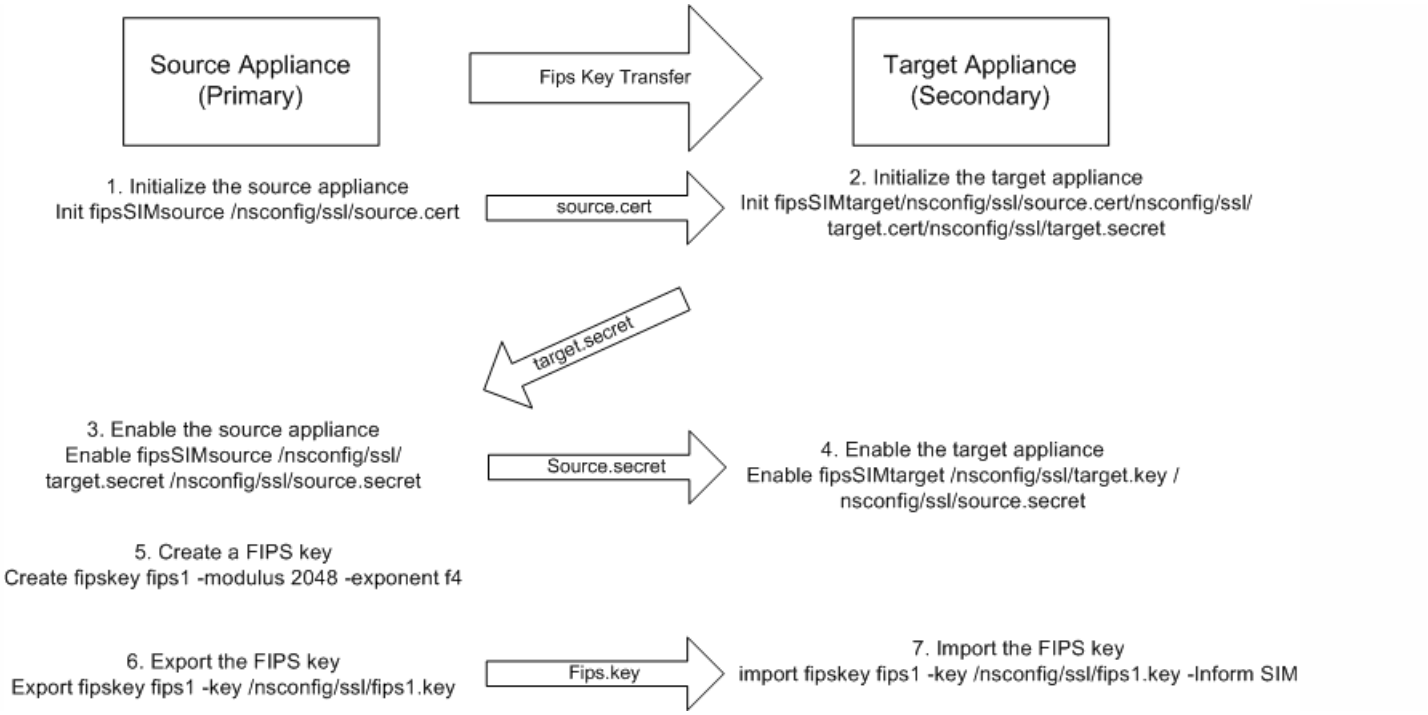
Copy this key into the hard disk of the target appliance.

On the target appliance

```
import fipskey fips1 -key /nsconfig/ssl/fips1.key
```

The following diagram summarizes the transfer process.

Figure 1. Transferring the FIPS Key-Summary



Resetting a Locked HSM

Nov 11, 2013

The HSM becomes locked (no longer operational) if you change the SO password, restart the appliance without saving the configuration, and make three unsuccessful attempts to change the password. This is a security measure for preventing unauthorized access attempts and changes to the HSM settings.

Important: To avoid this situation, save the configuration after initializing the HSM.
If the HSM is locked, you must reset the HSM and restart the appliance to restore the default passwords. You can then use the default passwords to access the HSM and configure it with new passwords. When finished, you must save the configuration and restart the appliance.

Caution: Do not reset the HSM unless it has become locked.

At the command prompt, type the following commands to reset and re-initialize a locked HSM:

- reset ssl fips
- reboot -warm
- set ssl fips -initHSM Level-2 <new SO password> <old SO password> <user password> [-hsmLabel <string>]
- save ns config
- reboot -warm

Example

```
reset fips
reboot -warm
set fips -initHSM Level-2 newsopin123 sopin123 userpin123 -hsmLabel NSFIPS
saveconfig
reboot -warm
```

Note: The SO and User passwords are the default passwords.

1. Navigate to Traffic Management > SSL > FIPS
2. In the details pane, on the FIPS Info tab, click Reset FIPS.
3. Configure the HSM, as described in Configuring the HSM.
4. In the details pane, click Save.

FIPS Approved Algorithms and Ciphers

Mar 21, 2012

The FIPS approved algorithms are:

Key-Exchange algorithms

- RSA

Cipher algorithms

- SSL3-DES-CBC3-SHA
- TLS1-AES-256-CBC-SHA
- TLS1-AES-128-CBC-SHA

Note: RC4 (ARC4) is not a FIPS-approved algorithm.
SSL virtual server is marked UP only when default ciphers (FIPS) are configured.

Troubleshooting

Jul 22, 2013

If the SSL feature does not work as expected after you have configured it, you can use some common tools to access NetScaler resources and diagnose the problem.

Updated: 2013-07-22

For best results, use the following resources to troubleshoot an SSL issue on a NetScaler appliance:

- The relevant ns.log file
- The latest ns.conf file
- The messages file
- The relevant newnslog file
- Trace files
- A copy of the certificate files, if possible
- A copy of the key file, if possible
- The error message, if any

In addition to the above resources, you can use the Wireshark application customized for the NetScaler trace files to expedite troubleshooting.

Updated: 2013-07-22

To troubleshoot an SSL issue, proceed as follows:

- Verify that the NetScaler appliance is licensed for SSL Offloading and load balancing.
- Verify that SSL Offloading and load balancing features are enabled on the appliance.
- Verify that the status of the SSL virtual server is not displayed as DOWN.
- Verify that the status of the service bound to the virtual server is not displayed as DOWN.
- Verify that a valid certificate is bound to the virtual server.
- Verify that the service is using an appropriate port, preferably port 443.

NetScaler Web 2.0 Push

Mar 19, 2012

Modern web applications, also referred to as web 2.0 applications, provide highly responsive interfaces that generate asynchronous updates that can impose an additional load on a server. Typically, asynchronous notifications are sent by using HTTP and server push techniques, such as long-polling and streaming response, which enable servers to push the notifications to clients. These techniques require the servers to maintain a large number of TCP/IP connections, which provides low latency but results in low bandwidth. As the number of clients increases, the servers are overloaded with connections kept open for each client. Further, the large number of connections terminating on the server requires kernel resources and memory for data structures like protocol control blocks, socket descriptors, and socket buffers.

With the NetScaler Web 2.0 push feature, you can use the NetScaler appliance as a proxy server to offload long-lived client TCP connections and maintain relatively fewer, reusable connections to the server. NetScaler Web 2.0 push is application agnostic, with the flexibility to work seamlessly with various technologies and configurations used for asynchronous messaging. It can be extended to co-exist with developing technologies, and it preserves backward compatibility. NetScaler Web 2.0 Push is also scalable, with support for multiple NetScaler appliances.

With the NetScaler Web 2.0 push feature, the NetScaler appliance multiplexes and manages the exchange of data reliably and securely, reducing the number of server-side connections across potentially millions of persistent client connections. For every HTTP, HTTPS, or SSL transaction, the appliance can de-link and rebalance the server farm to distribute client requests across multiple servers.

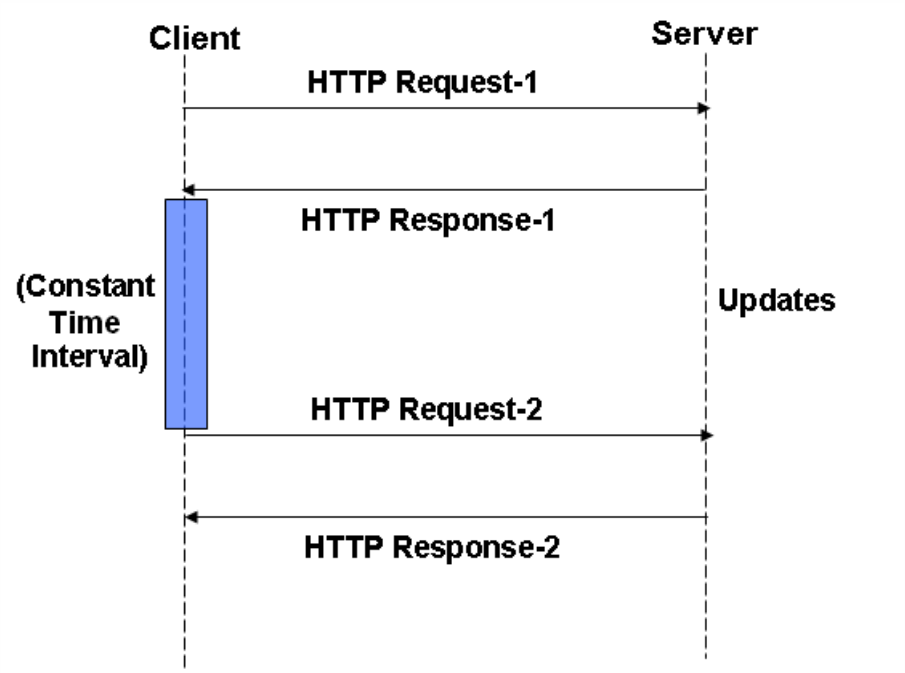
The NetScaler Web 2.0 Push feature reduces the number of server-side connections across millions of persistent client connections.

Web 2.0 Push Applications

Mar 19, 2012

With modern Web applications, termed broadly as Web 2.0 applications, servers use AJAX technologies such as polling to maintain up-to-date information about the client. Polling enables an AJAX application to periodically poll the server for updates. For example, a chat based application can poll a Web server every 10 seconds for any chat updates. To get such updates from the Web server, the client browser periodically opens a connection to the Web server.

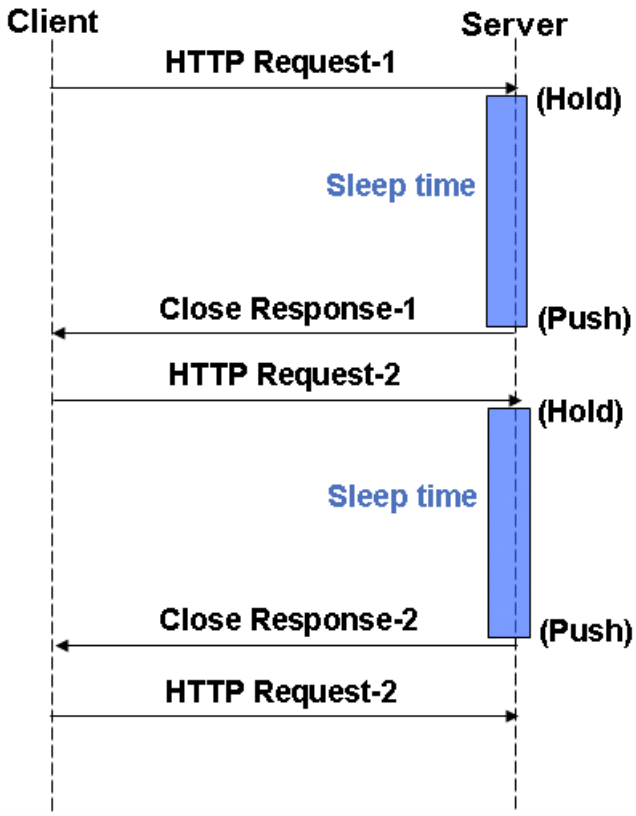
Figure 1. Polling Technique



Such frequent polling can overload the server. Also, if you deploy the AJAX application on a Web server with low resources and a large number of simultaneous users poll the server for updates, the network can become saturated, with significant degradation in the server performance. And if there is no update from the server, the client requests overload the server for a void response.

To avoid such problems, server push technology often uses a long polling technique. Long polling enables the client application to open a persistent connection to the server and wait for the server to push updates when available.

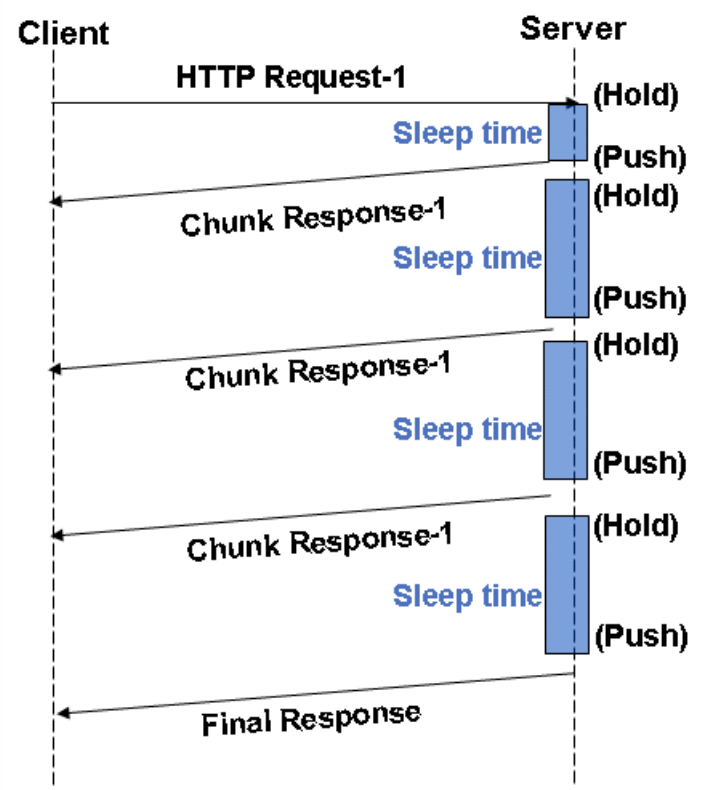
Figure 2. Long Polling Technique



If your server supports asynchronous request processing, long polling is a scalable technique. However, long polling can hold the server connections until updates are available. For example, if 1,000 AJAX applications open one long polled connection, 1,000 threads hold the server while waiting for updates.

Another technique, called HTTP streaming, is identical to the long polling technique except that the connection is not closed after the server pushes the updates. The AJAX application sends a single request and receives chunks of responses (partial responses) over the same connection. With HTTP streaming, the browsers and server do not open or close the connection. Therefore, HTTP streaming significantly reduces the network latency.

Figure 3. HTTP Streaming Technique



In HTTP streaming, as in long polling techniques, if the server frequently pushes updates, the performance of the network and the AJAX applications are significantly degraded and the client may lose the updates. If your AJAX application opens both long polling and HTTP streaming connections to the same Web server, other AJAX applications cannot open connections to the server, because the browser blocks such connections.

NetScaler Web 2.0 push uses connection labeling to overcome the limitations of long polling and HTTP streaming.

How Web 2.0 Push Works

Mar 19, 2012

The NetScaler Web 2.0 push feature enables the server to label a client connection and subsequently identify and send data over that labeled connection. With NetScaler Web 2.0 push enabled, the client first establishes a TCP/IP connection and connects to the NetScaler appliance. The appliance uses the configured load balancing method or content switching policy to select a Web server to which to open a connection and send the client request. The server interacts with the client and uses either authentication or a previously established cookie to identify the client.

When the NetScaler appliance receives a request with push enabled, it initiates the labeling protocol with the Web server. This protocol enables the Web server to label the connection and defer the response. The protocol also enables the server to process other requests without invoking push processing. The Web server (referred to as a *notification server*) uses the label to send updates to the client through the NetScaler appliance when the updates become available. Servers can choose to push multiple updates over a single TCP connection or open one connection per update.

Note: The set of Web servers that respond to requests from the NetScaler does not necessarily include the notification servers that push updates to client.
A central component of a NetScaler Web 2.0 push configuration is a push virtual server, which is a load balancing virtual server with service type PUSH or SSL_PUSH. The NetScaler appliance uses the push virtual server to expose the message push protocol to the Web servers. A server uses the protocol to push asynchronous messages to connected clients. A push virtual server exposes a simple REST interface for posting updates.

Important: For the NetScaler Web 2.0 Push feature to work correctly, you must configure the NetScaler appliance as a proxy for the traffic between the clients and servers. You can use multiple NetScaler appliances to scale up your connection management.
For each transaction, the NetScaler Web 2.0 push feature maintains a state machine, which manages the actions of the transaction. The state machine has the following states:

- Waiting for Request State (Q)-A connection has been established between the client and the NetScaler appliance. The appliance waits in this state until the client sends a request.
- Waiting for Server Response State (R) -A request has been received from the client and forwarded to a Web server. The appliance waits in this state for the server to respond.
- Waiting for Asynchronous Messages State (A) -The appliance is waiting for asynchronous messages that the notification servers push to the push virtual server.

Until the client establishes a connection with the NetScaler appliance's load balancing or content switching virtual server, the initial state of the transaction is Q. When the appliance receives a request, it forwards the request to the server, and the transaction moves to state R.

If the appliance receives a deferred response (also called a *labeled response*), the transaction moves to state A. In this state, if the appliance receives a push message through the message push protocol, it processes the message and forwards the message to the client. If this message is marked as the last message, the appliance closes the transaction and moves to state Q. If not, the transaction remains in state A.

The push virtual server can manage long-polling and streaming responses from the server. Each update that the server sends to the push virtual server has a flag (with query parameters) that indicates whether there are updates from the server. When the flag indicates that the updates from the server are unavailable, the NetScaler appliance performs one of the following functions:

1. If the client uses HTTP 1.1 protocol and multiple updates are received from the server, the appliance sends a chunked response to the client and appends a zero chunk to the final response. If the first response itself has the flag set, the content length itself is sent as the response.
2. If the client uses HTTP 1.0 protocol and multiple updates are received from the server, then just the contents of the chunked response or the body of the content length response is sent to the client and the connection is terminated. If the first response itself has the flag set then, the content length itself is sent as the response.

The appliance sends a content-length response regardless of which HTTP version the client uses. The connection-labeling and message-push protocols, which identify the client and the server connections, provide the basic functionality of the NetScaler Web 2.0 push feature.

Understanding NetScaler Web 2.0 Push Protocol

Mar 19, 2012

For the NetScaler Web 2.0 Push feature to work correctly, the NetScaler appliance must label the client connection and then identify and send the deferred response from the server over the labeled connection. For this purpose, the Web 2.0 push feature uses the connection labeling and the message push protocols.

The connection labeling protocol is used between the server and the NetScaler appliance to label the client connection. After a label is negotiated, the Web server includes the label in the update that is sent to the client.

The appliance forwards a request to the server after adding an X-NS-PUSHVSERVER header containing the IP address and port of the push virtual server. The server either responds to this request with an HTTP response or defers the response. If the server defers the response, it labels the connection with an X-NS-DEFERRABLE header, which indicates that the connection is deferred.

A policy configured on the load balancing or content switching virtual server enables the NetScaler appliance to extract the label from the response. The appliance uses the information in the label to send the push message (update) to the push virtual server, which sends the response on the corresponding client connection.

Note: For any update from the Web server, the NetScaler does not support rewrite and compression.
When a server receives a request that it is deferrable, it sends an HTTP 200 OK response with the X-NS-DEFERRABLE header, which indicates to the NetScaler appliance that the push feature should be applied to the request. The appliance removes the X-NS-DEFERRABLE header, sends the response to the client, and waits for updates. For example:

```
HTTP/1.1 200 OK
Date: Wed, 25 Aug 2010 18:22:47 GMT
Server: Apache/2.0.61 (FreeBSD) PHP/5.2.5 with Suhosin-Patch mod_ssl/2.0.61
OpenSSL/0.9.8e mod_perl/2.0.3 Perl/v5.8.8
X-NS-DEFERRABLE: YES
X-NS-SERVERLABEL: 04c2442bcb7c4b5f826d41a623e374e!
Content-Length: 0
Content-Type: text/plain;charset=UTF-8
```

The message push protocol is used between a notification server and the NetScaler appliance to enable the notification server to send a notification to a previously labeled client connection.

Web servers use the message push protocol to push asynchronous messages to connected clients. The push protocol is built as a REST interface, exposed through the push virtual server on the NetScaler appliance. The server connects to the push virtual server and sends a request to the appliance. The BODY of the request contains the payload to be sent to the client. Additionally, the request identifies the label for the target client connection and the last message of the response.

When the NetScaler appliance receives the deferred response from the server, it sends the response to the client as a single HTTP chunk and sends a 200 OK response with the XML information to the server. If the message is marked as the last message of the response, the NetScaler also closes the HTTP response on the server.

Note: If the NetScaler is aware of the content length, it may send a response specifying the Content-Length, instead of a chunk. This enables the NetScaler to manage both HTTP streaming and long-polling responses.

Notification from Server to Push Server

```
POST /CLIENT/V10/04c2442bcb7c4b5f826d41a623e374e!?MSG_END=0 HTTP/1.1
Host: 10.102.80.66:8080
Content-Length: 6
```

Response from Push vserver to the server

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="UTF-8"
Content-Length: 130
<?xml version="1.0" encoding="UTF-8"?>
<CLIENTINFO>
 <CLIENT ID="04c2442bcb7c4b5f826d41a623e374e!" INFO="SUCCESS" />
</CLIENTINFO>
```

Configuring Web 2.0 Push

Mar 19, 2012

To configure NetScaler Web 2.0 push, you must first enable the feature. Then, create a push virtual server and associate it with a load balancing or content switching virtual server. Once you have a working configuration, you can customize it to suit your deployment.

You can also monitor the Web 2.0 push configuration by viewing statistics about the push virtual server and the other entities, such as the load balancing or the content switching virtual servers, that are part of the configuration.

Enabling NetScaler Web 2.0 Push

Oct 15, 2013

You have to enable the NetScaler Web 2.0 push feature before you can use it. Before enabling the feature, you must have the appropriate license installed on the NetScaler appliance. With the feature disabled, you can configure NetScaler Web 2.0 push entities, such as the push virtual server, but the entities will not work until the feature is enabled.

At the command prompt, type:

enable ns feature push

If NetScaler Web 2.0 Push is not licensed or disabled, the push virtual server state is DOWN.

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Change advanced features.
3. In the Configure advanced features dialog box, select the NetScaler Push check box, and then click OK.
4. At the Enable/Disable Feature(s)? prompt, click Yes.

Creating a NetScaler Web 2.0 Push Virtual Server

Sep 02, 2013

A push virtual server enables the NetScaler appliance to multiplex and manage the exchange of data (server push) reliably, securely, and in a scalable manner. It enables the notification server to send a notification to a previously labeled client connection by using the message push protocol. The notification servers push the out-of-band updates to the push virtual server. When the clients access the load balancing or the content switching virtual servers, the push virtual server uses the labeling protocol to label the deferred clients.

You can add, modify, and remove push virtual servers, however, you cannot bind services to the push virtual server.

At a command prompt, type the following commands to create a push virtual server and verify the configuration:

- add lb vserver <name> <serviceType> <IPAddress> <Port>
- show lb vserver <name>

Example

```
add lb vserver Vserver-Push-1 PUSH 10.102.29.162 80
show lb vserver Vserver-Push-1
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, click Add.
3. In the Name, Port, and IP Address text boxes, type a name for the push virtual server, a port, and an IP address (for example, **Vserver-Push-1**, **80**, and **10.102.29.162**).
4. In Protocol, select either SSL_PUSH or PUSH.
5. Click Create, and then click Close. The push virtual server you created appears in the Load Balancing Virtual Servers pane.

To remove a push virtual server, use the rm lb vserver command that takes only the name parameter.

Configuring a Load Balancing or Content Switching Virtual Server

Oct 15, 2013

After creating a push virtual server, you need to associate it with the load balancing or content switching virtual servers. For details about creating a load balancing virtual server, see "Creating a Virtual Server." Also, for details about creating a content switching setup, see "Creating Content Switching Virtual Servers."

Once you have created the load balancing or content switching virtual servers, you must associate them with the push virtual server.

At the command prompt, type the following commands to configure a load balancing virtual server for NetScaler Web 2.0 push. To configure a content switching virtual server, replace set lb vserver with set cs vserver.

set lb vserver <name> <ServiceType> <IPAddress> <Port> -push (ENABLED | DISABLED) -pushVserver <PushVservername> -pushLabel <Expression> -pushMultiClients (YES | NO)

Examples

```
set lb vserver Vserver-LB-1 HTTP 10.102.29.161 80 -push ENABLED - pushVserver PushVserver1 -pushLabel "HTTP.RES.HEADER(\"NSLABEL\").VALUE(0)" –pushMultiClients
```

```
set cs vserver Vserver-CS-1 HTTP 10.102.29.161 80 -push ENABLED - pushVserver PushVserver1 -pushLabel "HTTP.RES.HEADER(\"NSLABEL\").VALUE(0)" –pushMultiClients
```

- To modify a virtual server, type the set lb vserver or set cs vserver command, the name of the virtual server, and the parameters to be changed, with their new values.
- To remove a virtual server, type the rm lb vserver or rm cs vserver command and the name of the load balancing or content switching virtual server.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure push virtual server (for example, Vserver-LB-1), and click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, on the Advanced tab, click the arrow next to Push to expand it. Then, in the Push VServer list, select the push virtual server and click OK.
   Note: To create a content switching virtual server for NetScaler Web 2.0 Push by using the configuration utility, in the navigation pane, expand Content Switching, click Virtual Servers, Then, perform steps 2 and 3.

Monitoring the Configuration

Sep 02, 2013

To monitor the NetScaler Web 2.0 push configuration, you need to view the statistics of the push virtual servers and load balancing entities. This is useful for troubleshooting.

For instructions on how to display statistics of load balancing entities, see "Load Balancing." Available statistics include labeled connections, push labeled connections, and deferred requests.

At the command prompt, type:

show lb vserver <PushVserverName>

Example

```
show lb vserver Vserver-Push-1
```

Customizing the NetScaler Web 2.0 Push Configuration

Mar 19, 2012

Once your basic Web 2.0 Push configuration is operational, you can customize it by setting a time-out value for idle client connections and configuring URL redirects.

Setting a Time-out Value for Idle Client Connections

Sep 02, 2013

Once a client connects to the push virtual server, you can configure the virtual server to close any idle client connections after a configured time period.

To configure a time-out value, use the cltTimeout parameter, which specifies the time, in seconds, after which the NetScaler appliance closes any idle client connections. The default value is 180sec for HTTP/SSL-based services and 9000sec for TCP-based services.

At the command prompt, type:

set lb vserver <PushVserverName> [-cltTimeout <secs>]

Example

```
set lb vserver Vserver-Push-1 -cltTimeout 100
```

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server for which you want to configure virtual server port insertion (for example, **Vserver-Push-1**), and then click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, click the Advanced tab.
4. In the Client Time-out (secs) text box, type the timeout value (for example, **100**).
5. Click OK.

Redirecting Client Requests to an Alternative URL

Sep 02, 2013

You can configure a URL to which to redirect HTTP or HTTPS client requests when the push virtual server is down or disabled. This URL can be a local or a remote link. The NetScaler appliance uses HTTP 302 redirect to redirect client requests.

Redirects can be absolute URLs or relative URLs. If the configured redirect URL contains an absolute URL, the HTTP redirect is sent to the configured location, regardless of the URL specified in the incoming HTTP request. If the configured redirect URL contains only a domain name (relative URL), the incoming URL is appended to the domain configured in the redirect URL.

The domain specified in the redirect URL must not be the same as the domain specified in the domain name argument of a content switching policy. If the same domain is specified in both arguments, the request is redirected continuously to the same unavailable virtual server in the NetScaler appliance, and the user cannot get the requested content.

At the command prompt, type:

set lb vserver <name> -redirectURL URLValue

Example

set lb vserver Vserver-Push-1 -redirectURL http://www.newdomain.com/mysite/maintenance

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the push virtual server for which you want to configure redirect URL (for example, Vserver-Push-1), and then click Open.
3. On the Advanced tab, in the Redirect URL text box, type the URL (for example, **http://www.newdomain.com/mysite/maintenance**).
4. Click OK.

Optimization

Aug 06, 2014

The NetScaler optimization features reduce transaction times between the clients and the servers, and they reduce bandwidth consumption. They also enhance server performance by offloading some tasks and making others more efficient.

Client Keep-Alive	Handles multiple requests on a single client connection. The client does not have to negotiate a new connection for each request to the server.
HTTP Compression	Compresses HTTP responses sent from the servers to compression-aware browsers. The smaller responses reduce download time and save bandwidth.
Integrated Caching	Stores responses to client requests. Subsequent requests for the same content are served from the NetScaler cache instead of being forwarded to the origin server.
Front End Optimization	Reduces the load and render time of web pages by simplifying and optimizing the content served to the client browser. Note: Supported from NetScaler 10.5 onwards.
Content Accelerator	Stores server responses on a Citrix ByteMobile T2100 appliance. Note: Supported from NetScaler 10.1 onwards.
SPDY (Speedy)	Acts as a SPDY gateway between clients and your servers, providing SPDY support without the need to configure/upgrade SPDY on the servers. Note: Supported from NetScaler 10.1 onwards.

Client Keep-Alive

Aug 06, 2014

The client keep-alive feature enables multiple client requests to be sent on a single client connection. This feature helps in a transaction management environment where typically the server closes the client connection after serving the response. The client then opens a new connection for each request and spends more time on the transaction.

Client keep-alive resolves this issue by keeping the connection between the client and the appliance (client-side connection) open even after the server closes the connection with the appliance. This allows sending multiple client requests using a single connection and saves the round trips associated with opening and closing a connection. Client keep-alive is most beneficial in SSL sessions.

Client keep-alive is also useful under either of the following conditions:

- When the server does not support client keep-alive.
- When the server supports client keep-alive but an application on the server does not support client keep-alive.

Note: Client keep-alive is applicable for HTTP and SSL traffic.
Client-keep alive can be configured globally to be able to handle all traffic. It can also be configured to be active only on specific services.

In client keep-alive environment, the configured services intercept the client traffic and the client request is directed to the origin server. The server sends the response and closes the connection between the server and the appliance. If a "Connection: Close" header is present in the server response, the appliance corrupts this header in the client-side response, and the client-side connection is kept open. As a result, the client does not have to open a new connection for the next request; instead, the connection to the server is reopened.

Note: If a server sends back two "Connection: Close" headers, only one is edited. This results in significant delays on the client rendering of the object because a client does not assume that the object has been delivered completely until the connection is actually closed.

Updated: 2014-08-12

Client keep-alive, by default, is disabled on the NetScaler, both globally and at service level. Therefore, you must enable the feature at the required scope.

Note: If you enable client keep-alive globally, it is enabled for all services, regardless of whether you enable it at the service level.
Additionally, if required, you can configure some HTTP parameters to specify the maximum number of HTTP connections retained in the connection reuse pool, enable connection multiplexing, and enable persistence Etag.

Note: When Persistent ETag is enabled, the ETag header includes information about the server that served the content. This ensures that cache validation conditional requests or browser requests, for that content, always reaches the same server.

To configure client keep-alive by using the command line interface

At the command prompt, do the following:

1. Enable client keep-alive on the NetScaler.

- At global level
  enable ns mode cka
- At service level
  set service <name> -CKA YES

  Note: Client keep-alive can be enabled only for HTTP and SSL services.

2. Configure the required HTTP parameters on the HTTP profile that is bound to the service(s).
   set ns httpProfile <name> -maxReusePool <value> -conMultiplex ENABLED -persistentETag ENABLED

   Note: Configure these parameters on the nshttp_default_profile HTTP profile, to make them available globally.

To configure client keep-alive by using the configuration utility

1. Enable client keep-alive on the NetScaler.
   - At global level
     Navigate to System > Settings, click Configure Modes and select Client side Keep Alive.
   - At service level
     Navigate to Traffic Management > Load Balancing > Services, and select the required service. In the Settings grouping, enable Client Keep-Alive.
2. Configure the required HTTP parameters on the HTTP profile that is bound to the service(s).
   Navigate to System > Profiles, and on HTTP Profiles tab, select the required profile and update the required HTTP parameters.

HTTP Compression

May 20, 2015

For websites with compressible content, the NetScaler HTTP compression feature implements lossless compression to alleviate latency, long download times, and other network-performance problems by compressing the HTTP responses sent from servers to compression-aware browsers. You can improve server performance by offloading the computationally intensive compression task from your servers to the NetScaler appliance.

The following table describes the capabilities of the HTTP compression feature:

Functionality	Description
Compression Ratio	Compression ratio depends on the types of files in the responses, but is always significant, noticeably reducing amount of data transmitted over the network.
Browser Awareness	NetScaler serves compressed data to compression aware browsers only, reducing the transaction time between the client and the server. Most modern web browsers support HTTP compression.
Compression blocking	You can define content filters to selectively block compression by applying built-in actions.
Compression Caching	With the integrated caching feature enabled, subsequent requests for the same content are served from the local cache, reducing the number of round trips to the server and improving transaction times.
HTTPS Support	Compression is particularly useful on SSL connections, because it reduces the amount of content that has to be encrypted, either on the server or by the NetScaler appliance, and decrypted by the client.
Intelligent Response Filtering	The NetScaler compression engine intelligently filters server responses on the basis of defined compression parameters. For example, the compression engine detects zero-content-length responses and compressed responses and does not compress them. The detection of compressed responses enables origin sites to use server-based compression in conjunction with the NetScaler compression feature.
Compression Switching	The NetScaler appliance transparently directs requests from compression aware clients to compression capable servers, so that responses to those clients are compressed, and responses to other clients are not delayed by compression processing.

How Compression Works

A NetScaler ADC can compress both static and dynamically generated data. It applies the GZIP or the DEFLATE compression algorithm to remove extraneous and repetitive information from the server responses and represent the original information in a more compact and efficient format. This compressed data is sent to the client's browser and

uncompressed as determined by the browser's supported algorithm or algorithms (GZIP or DEFLATE).

NetScaler compression treats static and dynamic content differently.

- Static files are compressed only once, and a compressed copy is stored in local memory. Subsequent client requests for cached files are serviced from that memory.
- Dynamic pages are dynamically created each time a client requests them.

When a client sends a request to the server:

1. The client request arrives at the NetScaler ADC. The ADC examines the headers and stores information about what kind of compression, if any, the browser supports.
2. The ADC forwards the request to the server and receives the response.
3. The NetScaler compression engine examines the server response for compressibility by matching it against policies.
4. If the response matches a policy associated with a compression action, and the client browser supports a compression algorithm specified by the action, the NetScaler ADC applies the algorithm and sends the compressed response to the client browser.
5. The client applies the supported compression algorithm to decompress the response.

Configuring HTTP Compression

Nov 03, 2015

By default, compression is disabled on the NetScaler ADC. You must enable the feature before configuring it. If the feature is enabled, the ADC compresses server requests specified by compression policies.

To configure HTTP compression, do the following:

- Configure compression actions
- Configure compression policies
- Bind the compression policies to global bind points or to virtual servers
- Optionally, configure global compression parameters

Enabling HTTP Compression

Compression can be enabled for HTTP and SSL services only. You can enable it globally, so that it applies to all HTTP and SSL services, or you can enable it just for specific services.

To enable compression by using the command line interface

At the command prompt, enter one of the following commands to enable compression globally or for a specific service:

- enable ns feature cmp
  OR
- set service <name> -CMP YES

To configure compression by using the configuration utility

Do one of the following:

- To enable compression globally, navigate to **System** > **Settings**, click **Configure Basic Features**, and select **HTTP Compression**.
- To enable compression for a specific service, navigate to **Traffic Management** > **Load Balancing** > **Services**, select the service, and click **Edit**. In the **Settings** group, click the pencil icon and enable **Compression**.

Configuring a Compression Action

A compression action specifies the action to take when a request or response matches the rule (expression) in the policy with which the action is associated. For example, you can configure a compression policy that identifies requests that will be sent to a particular server, and associate the policy with an action that compresses the server's response.

There are four built-in compression actions:

- COMPRESS: Uses the GZIP algorithm to compress data from browsers that support either GZIP or both GZIP and DEFLATE. Uses the DEFLATE algorithm to compress data from browsers that support only the DEFLATE algorithm. If the browser does not support either algorithm, the browser's response is not compressed.
- NOCOMPRESS: Does not compress data.
- GZIP: Uses the GZIP algorithm to compress data for browsers that support GZIP compression. If the browser does not support the GZIP algorithm, the browser's response is not compressed.
- DEFLATE: Uses the DEFLATE algorithm to compress data for browsers that support the DEFLATE algorithm. If the

browser does not support the DEFLATE algorithm, the browser's response is not compressed. After creating an action, you associate the action with one or more compression policies.

To create a compression action by using the command line interface

At the command prompt, enter the following command to create a compression action:

add cmp action <name> <cmpType>

To create a compression action by using the configuration utility

Navigate to **Optimization** > **HTTP Compression** > **Actions** , click **Add** , and create a compression action to specify the type of compression to be performed on the HTTP response.

Configuring a Compression Policy

A compression policy contains a rule, which is a logical expression that enables the NetScaler appliance to identify the traffic that should be compressed.

When the NetScaler ADC receives an HTTP response from a server, it evaluates the built-in compression policies and any custom compression policies to determine whether to compress the response and, if so, the type of compression to apply. Priorities assigned to the policies determine the order in which the policies are matched against the requests.

The following table lists the built-in HTTP compression policies. These policies are activated globally when you enable compression.

Built-in Classic or Default Syntax Policy	**Description**
ns_nocmp_mozilla_47 ns_adv_nocmp_mozilla_47	Prevents compression of CSS files when a request is sent from a Mozilla 4.7 browser.
ns_cmp_mscss ns_adv_cmp_mscss	Compresses CSS files when the request is sent from a Microsoft Internet Explorer browser.
ns_cmp_msapp ns_adv_cmp_msapp	Compresses files that are generated by the following applications:s • Microsoft Office Word • Microsoft Office Excel • Microsoft Office PowerPoint
ns_cmp_content_type ns_adv_cmp_content_type	Compresses data when the response containsContent-Type header and contains text.
ns_nocmp_xml_ie	Prevents compression when a request is sent, from a Microsoft Internet Explorer browser and the response contains a Content-Type header and contains text or xml.

Built-in Classic or Default Syntax Policy	Description
ns_adv_nocmp_xml_ie	

To create a compression policy by using the command line interface

At the command prompt, enter the following command to create a compression policy:

add cmp policy <name> -rule <expression> -resAction <string>

To create a compression policy by using the configuration utility

Navigate to **Optimization** > **HTTP Compression** > **Policies** , click **Add** , and create a compression policy by specifying the condition and the corresponding action to be executed.

Binding a Compression Policy

To put a compression policy into effect, you must bind it either globally, so that it applies to all traffic that flows through the NetScaler ADC, or to a specific virtual server, so that the policy applies only to requests whose destination is the VIP address of that virtual server.

When you bind a policy, you assign it a priority. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer.

To bind a compression policy by using the command line interface

At the command prompt, enter one of the following commands to bind a compression policy globally or to a specific virtual server:

- bind cmp global <policyName> [-priority <positive_integer>] [-state (ENABLED | DISABLED)]...
- bind lb vserver <vserverName> -policyName <policyName> -priority <positive_integer>. Repeat this command for each virtual server to which you want to bind the compression policy.

To bind a compression policy by using the configuration utility

Do one of the following:

- At global level Navigate to **Optimization** > **HTTP Compression** > **Policies**, click **Policy Manager** and bind the required policies by specifying the relevant **Bind Point** and **Connection Type** (Request/Response).
- At virtual server level
  - For load balancing virtual server, Navigate to **Traffic Management** > **Load Balancing** > **Virtual Servers**, select the required virtual server, click **Policies**, and bind the relevant policy.
  - For content switching **virtual server**, Navigate to **Traffic Management** > **Content Switching** > **Virtual Servers**, select the required virtual server, click **Policies**, and bind the relevant policy.

Setting the Global Compression Parameters for Optimal Performance

Many users accept the default values for the global compression parameters, but you might be able provide more effective compression by customizing these settings.

The following table describes the compression parameters that you can set on the NetScaler ADC.

Compression Parameters	Description
Quantum size	Size, in KB, of the buffer maintained for accumulating server responses. The responses are compressed when the buffer size exceeds this value. For example, if you set the quantum size to 50 KB, the NetScaler ADC compresses the buffer's contents when its size becomes larger than 50 KB. Minimum value: 1. Maximum value: 63488. Default: 57344.
Compression level	Level of compression to apply to server responses. Possible values: Best Speed, Best Compression, optimal.
Minimum HTTP response size	Minimum size, in bytes, of an HTTP response that is compressed. Responses smaller than the value specified by this parameter are sent without being compressed.
Bypass compression on CPU usage	NetScaler CPU usage, as a percentage, at or above which no compression is done. Default: 100.
Policy Type*	Type of policies used for compression. Possible values: Classic, Default Syntax. Default: Classic.
Allow Server-side compression	Allow servers to send compressed data to the NetScaler ADC.
Compress push packet	Upon receipt of a packet with a TCP PUSH flag, compress the accumulated packets immediately, without waiting for the quantum buffer to be filled.
External Cache	Issue a private response directive indicating that the response message is intended for a single user and must not be cached by a shared or proxy cache.

To configure global compression parameters by using the command line interface

At the command prompt, enter the following command to configure compression parameters that apply globally:

set cmp parameter -cmpLevel <cmpLevel> -quantumSize <integer>

Note: Vary header parameters are available from NetScaler 10.5 onwards.

To configure global compression parameters by using the configuration utility

Navigate to **Optimization** > **HTTP Compression** , click **Change Compression Settings** , and set the relevant parameters.

Evaluating Your Compression Configuration

Apr 20, 2015

You can view the compression statistics in the dashboard utility or in an SNMP monitor. The dashboard utility displays summary and detailed statistics in a tabular and graphic format.

Optionally, you can also view statistics for a compression policy, including the number of hits that the policy counter increments during the policy based compression.

Note:

- For more information about the statistics and charts, see the Dashboard help on the Citrix NetScaler appliance.
- For more information about SNMP, see SNMP.

To View Compression Statistics by Using the Command Line Interface

At the command prompt, enter the following commands to display the compression statistics:

1. To display compression statistics summary
   stat cmp
   Note: The stat cmp policy command displays statistics for default syntax compression policies only.
2. To display compression policy hits and details
   show cmp policy <name>
3. To display detailed compression statistics
   stat cmp -detail

To View Compression Statistics by Using the Dashboard

In the Dashboard utility, you can display the following types of compression statistics:

- Select Compression to display a summary of the compression statistics.
- To display detailed compression statistics by protocol type, click the Details
- To display the rate of requests processed by the compression feature, click the Graphical View tab.

To View Compression Statistics by Using SNMP

You can view the following compression statistics by using the SNMP network management application.

- Number of compression requests (OID: 1.3.6.1.4.1.5951.4.1.1.50.1)
- Number of compressed bytes transmitted (OID: 1.3.6.1.4.1.5951.4.1.1.50.2)
- Number of compressible bytes received (OID: 1.3.6.1.4.1.5951.4.1.1.50.3)
- Number of compressible packets transmitted (OID: 1.3.6.1.4.1.5951.4.1.1.50.4)
- Number of compressible packets received (OID: 1.3.6.1.4.1.5951.4.1.1.50.5)
- Ratio of compressible data received and compressed data transmitted (OID: 1.3.6.1.4.1.5951.4.1.1.50.6)
- Ratio of total data received to total data transmitted (OID: 1.3.6.1.4.1.5951.4.1.1.50.7)

To View Additional Compression Statistics by Using the Configuration Utility

1. To display HTTP compression statistics:
   Navigate toOptimization > HTTP Compression and click Statistics.
2. To display statistics of a compression policy
   Navigate to Optimization > HTTP Compression > Policies> select the policy, and click Statistics.
3. To display statistics of a compression policy label

Navigate to Optimization > HTTP Compression > Policies> select a policy label, and click Statistics.

Offloading HTTP Compression to the NetScaler ADC

Nov 04, 2015

Performing compression on a server can affect the server's performance. A NetScaler ADC placed in front of your web servers and configured for HTTP compression offloads compression of both static and dynamic content, saving server CPU cycles and resources.

You can offload compression from the Web servers in either of two ways:

- Disable compression on the web servers, enable the NetScaler Compression feature at a global level, and configure services for compression.
- Leave the compression feature enabled on the web servers and configure the NetScaler appliance to remove the "Accept Encoding" header from all HTTP client requests. The servers then send uncompressed responses. The NetScaler ADC compresses the server responses before sending them to the clients.

Note: The second option does not work if the servers automatically compress all responses. The NetScaler ADC does not attempt to compress a response that is already compressed.

The Servercmp parameter enables the Netscaler appliance to handle offload HTTP compression. By default, this parameter is set ON for the server to send compressed data to the Netscaler appliance. To offload HTTP compression, you need to set the servercmp parameter to OFF.

At the command prompt, enter the following commands:

set service <service name> -CMP YES

Repeat this command for each service for which you want to enable compression.

show service <service name>

Repeat this command for each service, to verify that compression is enabled.

Save config

set cmp parameter –serverCmp OFF

Integrated Caching

Oct 28, 2013

The integrated cache provides in-memory storage on the Citrix NetScaler appliance and serves Web content to users without requiring a round trip to an origin server. For static content, the integrated cache requires little initial setup. After you enable the integrated cache feature and perform basic setup (for example, determining the amount of NetScaler appliance memory the cache is permitted to use), the integrated cache uses built-in policies to store and serve specific types of static content, including simple Web pages and image files. You can also configure the integrated cache to store and serve dynamic content that is usually marked as non-cacheable by Web and application servers (for example, database records and stock quotes).

When a request or response matches the rule (logical expression) specified in a built-in policy or a policy that you have created, the NetScaler appliance performs the action associated with the policy. By default, all policies store cached objects in and retrieve them from the Default content group, but you can create your own content groups for different types of content.

To enable the NetScaler appliance to find cached objects in a content group, you can configure selectors, which match cached objects against expressions, or you can specify parameters for finding objects in the content group. If you use selectors (which Citrix recommends), configure them first, so that you can specify selectors when you configure content groups. Next, set up any content groups that you want to add, so that they are available when you configure the policies. To complete the initial configuration, create policy banks by binding each policy to a global bind point or a virtual server, or to a label that can be called from other policy banks.

You can tune the performance of the integrated cache, using methods such as pre-loading cached objects before they are scheduled to expire. To manage the handling of cached data once it leaves the NetScaler appliance, you can configure caching-related headers that are inserted into responses. The integrated cache can also act as a forward proxy for other cache servers.

Note: Integrated caching requires some familiarity with HTTP requests and responses. For information about the structure of HTTP data, see *Live HTTP Headers* at "http://livehttpheaders.mozdev.org/."

How the Integrated Cache Works

Jun 03, 2014

The integrated cache monitors HTTP and SQL requests that flow through the Citrix NetScaler appliance and compares the requests with stored policies. Depending on the outcome, the integrated cache feature either searches the cache for the response or forwards the request to the origin server. For HTTP requests, the integrated cache feature can also serve partial content from the cache in response to single byte-range and multi-part byte-range requests.

Cached data can be compressed if the client accepts compressed content. You can configure expiration times for a content group, and you can selectively expire entries in a content group.

Data that is served from the integrated cache is a cache hit, and data served from the origin is a cache miss, as described in the following table.

Table 1. Cache Hits and Misses

Transaction Type	Specifies
Cache Hit	Responses that the NetScaler appliance serves from the cache, including: • Static objects, for example, image files and static Web pages • 200 OK pages • 203 Non-Authoritative Response pages • 300 Multiple Choices pages • 301 Moved Permanently pages • 302 Found pages • 304 Not Modified pages These responses are known as positive responses. The NetScaler appliance also caches the following negative responses: • 307 Temporary Redirect pages • 403 Forbidden pages • 404 Not Found pages • 410 Gone pages To further improve performance, you can configure the NetScaler appliance to cache additional types of content.
Storable Cache Miss	For a storable cache miss, the NetScaler appliance fetches the response from the origin server, and stores the response in the cache before serving it to the client.
Non-Storable Cache Miss	A non-storable cache miss is inappropriate for caching. By default, any response that contains the following status codes is a non-storable cache miss:

Transaction Type	**Specifies** • 201, 202, 204, 205, 206 status codes • All 4xx codes, except 403, 404 and 410 • 5xx status codes

Note: To integrate dynamic caching with your application infrastructure, use the XML API to issue cache commands remotely. For example, you can configure triggers that expire cached responses when a database table is updated.
To ensure the synchronization of cached responses with the data on the origin server, you configure expiration methods. When the NetScaler appliance receives a request that matches an expired response, it refreshes the response from the origin server.

Note: Citrix recommends that you synchronize the times on the NetScaler appliance and the back-end server(s).

Example of Dynamic Caching

Sep 11, 2014

Dynamic caching evaluates HTTP requests and responses based on parameter-value pairs, strings, string patterns, or other data. For example, suppose that a user searches for Bug 31231 in a bug reporting application. The browser sends the following request on the user's behalf:

```
GET /mybugreportingsystem/mybugreport.dll?IssuePage&RecordId=31231&Template=view&TableId=1000
Host: mycompany.net
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9) Gecko/2008052906 Firefox/3.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
. . .
```

In this example, GET requests for this bug reporting application always contain the following parameters:

- IssuePage
- RecordID
- Template
- TableId

GET requests do not update or alter the data, so you can configure these parameters in caching policies and selectors, as follows:

- You configure a caching policy that looks for the string mybugreportingsystem and the GET method in HTTP requests. This policy directs matching requests to a content group for bugs.
- In the content group for bugs, you configure a hit selector that matches various parameter-value pairs, including IssuePage, RecordID, and so on.

Note that a browser can send multiple GET requests based on one user action. The following is a series of three separate GET requests that a browser issues when a user searches for a bug based on a bug ID.

```
GET /mybugreportingsystem/mybugreport.dll?IssuePage&RecordId=31231&Template=view&TableId=1000
GET /mybugreportingsystem/mybugreport.dll?IssuePage&Template=viewbtns&RecordId=31231&TableId=1000
GET /mybugreportingsystem/mybugreport.dll?IssuePage&Template=viewbody&RecordId=31231&tableid=1000
```

To fulfill these requests, multiple responses are sent to the user's browser, and the Web page that the user sees is an assembly of the responses.

If a user updates a bug report, the corresponding responses in the cache should be refreshed with data from the origin server. The bug reporting application issues HTTP POST requests when a user updates a bug report. In this example, you configure the following to ensure that POST requests trigger invalidation in the cache:

- A request-time invalidation policy that looks for the string mybugreportingsystem and the POST HTTP request method, and directs matching requests to the content group for bug reports.
- An invalidation selector for the content group for bug reports that expires cached content based on the RecordID parameter. This parameter appears in all of the responses, so the invalidation selector can expire all relevant items in the cache.

The following excerpt shows a POST request that updates the sample bug report.

```
POST /mybugreportingsystem/mybugreport.dll?TransitionForm HTTP/1.1\r\n
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0) Opera 7.23 [en]\r\n
Host: mybugreportingsystem\r\n
Cookie:ttSearch.134=%23options%3Afalse%23active%23owner%3Afalse%23unowned%3Afalse%23submitter%3Afalse%23incsub%3Atrue;
Cookie2: $Version=1\r\n
. . .
\r\n
ProjectId=2&RecordId=31231&TableId=1000&TransitionId=1&Action=Update&CopyProjectId=0&ReloadForm=0&State=&RecordLockId=49873+issues+in+HTTP&F43. .
```

When the Citrix NetScaler appliance receives this request, it does the following:

- Matches the request with an invalidation policy.
- Finds the content group that is named in the policy.
- Applies the invalidation selector for this content group and expires all responses that match RecordID=31231.

When a user issues a new request for this bug report, the NetScaler appliance goes to the origin server for updated copies of all the responses that are associated with the report instance, stores the responses in the content group, and serves them to the user's browser, which reassembles the report and displays it.

Setting Up the Integrated Cache

May 20, 2015

To use the integrated cache, you must install the license and enable the feature. After you enable the integrated cache, the Citrix® NetScaler® appliance automatically caches static objects as specified by built-in policies and generates statistics on cache behavior. (Built-in policies have an underscore in the initial position of the policy name.)

Even if the built-in policies are adequate for your situation, you might want to modify the global attributes. For example, you might want to modify the amount of NetScaler appliance memory allocated to the integrated cache.

If you would like to observe cache operation before changing settings, see "Displaying Cached Objects and Cache Statistics."

Note: The NetScaler cache is an in-memory store that is purged when you restart the appliance.
This section includes the following details:

- Installing the Integrated Cache License
- Enabling Integrated Caching
- Configuring Global Attributes for Caching
- Built-in Content Group, Pattern Set, and Policies for the Integrated Cache

Installing the Integrated Cache License

Updated: 2013-10-28

An integrated cache license is required. For information about licenses, see information about obtaining NetScaler licenses at " http://support.citrix.com/article/ctx121062."

To install the license for the Integrated Caching feature

1. Obtain a license code from Citrix, go to the command line interface, and log in.
2. At the command line interface, copy the license file to the /nsconfig/license folder.
3. Reboot the NetScaler appliance by using the following command:

```
reboot
```

Enabling Integrated Caching

Updated: 2015-05-20

When you enable integrated caching, the NetScaler appliance begins caching server responses. If you have not configured any policies or content groups, the built in policies store cached objects in the Default content group.

To enable integrated caching by using the command line interface

At the command prompt, type one of the following commands to enable or disable integrated caching:

```
enable ns feature IC
```

To enable integrated caching by using the configuration utility

Navigate to System > Settings, click Configure Basic Features, and select Integrated Caching.

Configuring Global Attributes for Caching

Updated: 2014-08-08

Global attributes apply to all cached data. You can specify the amount of NetScaler memory allocated to the integrated cache, Via header insertion, a criterion for verifying that a cached object should be served, the maximum length of a POST body permitted in the cache, whether to bypass policy evaluation for HTTP GET requests, and an action to take when a policy cannot be evaluated.

The cache memory capacity is limited only by the memory of the hardware appliance. Also, any packet engine (the central distribution hub of all incoming TCP requests) in the nCore NetScaler appliance is aware of objects cached by other packet engines in the nCore NetScaler appliance.

Note that the default global memory limit is 0. Therefore, even if Integrated Caching is enabled, the NetScaler appliance does not cache any objects. You must explicitly set the global memory limit when integrated caching is enabled.

You can modify the global memory limit configured for caching objects. However, when you update the global memory limit to a value lower than the existing value (for example, from 10 GB to 4 GB), if a higher amount of memory (greater than 4 GB) is already being used to cache objects, the NetScaler continues using that amount of memory.

This means that even though the integrated caching limit is configured to some value, the actual limit used can be higher. This excessive memory is however released when the objects are removed from cache.

The output of the show cache parameter command indicates the configured value (Memory Usage limit) and the actual value being used (Memory usage limit (active value)).

To configure global settings for caching by using the command line interface

At the command prompt, type:

```
set cache parameter [-memLimit <MBytes>] [-via <string>] [-verifyUsing <criterion>] [-maxPostLen <positiveInteger>] [-prefetchMaxPending <positiveInteger>] [-enableBypass (YES | NO)] [-undefAction (NOCACHE | RESET)]
```

To configure global settings for caching by using the configuration utility

Navigate to Optimization > Integrated Caching, click Change Cache Settings, and configure the global settings for caching.

Built-in Content Group, Pattern Set, and Policies for the Integrated Cache

Updated: 2013-08-23

The Citrix NetScaler appliance includes a built-in integrated caching configuration that you can use for caching content. The configuration consists of a content group called ctx_cg_poc, a pattern set called ctx_file_extensions, and a set of integrated cache policies. In the content group ctx_cg_poc, only objects that are 500 KB or smaller are cached. The content is cached for 86000 seconds, and the memory limit for the content group is 512 MB. The pattern set is an indexed array of common file extensions for file-type matching.

The following table lists the built-in integrated caching policies. By default, the policies are not bound to any bind point. You must bind them to a bind point if you want the NetScaler appliance to evaluate traffic against the policies. The policies cache objects in the ctx_cg_poc content group.

Table 1. Built-in Integrated Caching Policies

Integrated caching policy name	Policy rule	Policy action	Description
ctx_images	HTTP.REQ.URL.SET_TEXT_MODE(IGNORECASE).CONTAINS_INDEX(\"ctx_file_extensions\").BETWEEN(101,150)	CACHE	Check whether the requested file is an image file (by comparing the extension of the requested object with the extensions in ctx_file_extensions, indexes 101 through 150) and then cache the file.
ctx_web_css	HTTP.REQ.URL.ENDSWITH(\".css\")	CACHE	Cache all cascading style sheet content.
ctx_doc_pdf	HTTP.REQ.URL.ENDSWITH(\".pdf\")	CACHE	Cache all PDF content.
ctx_web_JavaScript	HTTP.REQ.URL.ENDSWITH(\".js\")	CACHE	Cache all JSS content.
ctx_web_JavaScript-Res	HTTP.RES.HEADER(\"Content-Type\").CONTAINS(\"application/x-javascript\")	CACHE	Cache all JavaScript content.
ctx_NOCACHE_Cleanup	TRUE	NOCACHE	Do not cache any content.

Configuring Selectors and Basic Content Groups

Aug 26, 2013

You can configure selectors and apply them to content groups. When you add a selector to one or more content groups, you specify whether the selector is to be used for identifying cache hits or identifying cached objects to be invalidated (expired). Selectors are optional. Alternatively, you can configure content groups to use hit parameters and invalidation parameters. However, Citrix recommends that you configure selectors.

After configuring selectors, or deciding to use parameters instead, you are ready to set up a basic content group. After creating the basic content group, you need to decide how objects should be expired from the cache, and configure cache expiration. You can further modify the cache as described in "Improving Cache Performance" and "Configuring Cookies, Headers, and Polling", but you might first want to configure caching policies.

Note: Content group parameters and selectors are used only at request time, and you typically associate them with policies that use MAY_CACHE or MAY_NOCACHE actions.

Advantages of Selectors

Oct 28, 2013

A selector is a filter that locates particular objects in a content group. If you do not configure a selector, the Citrix® NetScaler® appliance looks for an exact match in the content group. This can lead to multiple copies of the same object residing in a content group. For example, a content group that does not have a selector may need to store URLs for host1.domain.com\mypage.htm, host2.domain.com\mypage.htm, and host3.domain.com\mypage.htm. In contrast, a selector can match just the URL (mypage.html, using the expression http.req.url) and the domain (.com, using the expression http.req.hostname.domain), allowing the requests to be satisfied by the same URL.

Selector expressions can perform simple matching of parameters (for example, to find objects that match a few query string parameters and their values). A selector expression can use Boolean logic, arithmetic operations, and combinations of attributes to identify objects (for example, segments of a URL stem, a query string, a string in a POST request body, a string in an HTTP header, a cookie). Selectors can also perform programmatic functions to analyze information in a request. For example, a selector can extract text in a POST body, convert the text into a list, and extract a specific item from the list.

For more information about expressions and what you can specify in an expression, see "Policies and Expressions."

Using Parameters Instead of Selectors

Jul 10, 2013

Although Citrix recommends the use of selectors with a content group, you can instead configure hit parameters and invalidation parameters. For example, suppose that you configure three hit parameters in a content group for bug reports: BugID, Issuer, and Assignee. If a request contains BugID=456, with Issuer=RohitV and Assignee=Robert, the NetScaler appliance can serve responses that match these parameter-value pairs.

Invalidation parameters in a content group expire cached entries. For example, suppose that BugID is ab invalidation parameter and a user issues a POST request to update a bug report. An invalidation policy directs the request to this content group, and the invalidation parameter for the content group expires all cached responses that match the BugID value. (The next time a user issues a GET request for this report, a caching policy can enable the NetScaler appliance to refresh the cached entry for the report from the origin server.)

Note that the same parameter can be used as a hit parameter or an invalidation parameter.

Content groups extract request parameters in the following order:

- URL query
- POST body
- Cookie header

After the first occurrence of a parameter, regardless of where it occurred in the request, all its subsequent occurrences are ignored. For example, if a parameter exists both in the URL query and in the POST body, only the one in the URL query is considered.

If you decide to use hit and invalidation parameters for a content group, configure the parameters when you configure the content group.

Note: Citrix recommends that you use selectors rather than parameterized content groups, because selectors are more flexible and can be adapted to more types of data.

Configuring a Selector

Aug 07, 2014

A content group can use a hit selector to retrieve cache hits or use an invalidation selector to expired cached objects and fetch new ones from the origin server.

A selector contains a name and a logical expression, called an *advanced expression*.

For more information about advanced expressions, see "Policies and Expressions."

To configure a selector, you assign it a name and enter one or more expressions. As a best practice, a selector expression should include the URL stem and host, unless there is a strong reason to omit them.

To configure a selector by using the command line interface

At the command prompt, type:

add cache selector <selectorName> (<rule> ...)

For information about configuring the expression or expressions, see "To configure a selector expression by using the command line interface."

Examples

```
>add cache selector product_selector "http.req.url.query.value(\"ProductId\")" "http.req.url.query.value(\"BatchNum\")" "http.req.url.query.value(\"depotLocation\")"
> add cache selector batch_selector "http.req.url.query.value(\"ProductId\")" "http.req.url.query.value(\"BatchId\")" "http.req.url.query.value(\"depotLocation\")"
> add cache selector product_id_selector "http.req.url.query.value(\"ProductId\")"
> add cache selector batchnum_selector "http.req.url.query.value(\"BatchNum\")" "http.req.url.query.value(\"depotLocation\")"
> add cache selector batchid_selector "http.req.url.query.value(\"depotLocation\")" "http.req.url.query.value(\"BatchId\")"
```

To configure a selector by using the configuration utility

Navigate to Optimization > Integrated Caching > Cache Selectors, and add the cache selector.

About Content Groups

Mar 16, 2012

A content group is a container for cached objects that can be served in a response. When you first enable the integrated cache, cacheable objects are stored in a content group named Default. You can create new content groups that have unique properties. For example, you can define separate content groups for image data, bug reports, and stock quotes, and you can configure the stock quote content group to be refreshed more often than the other groups.

You can configure expiration of an entire content group or selected entries in a content group.

The data in a content group can be static or dynamic, as follows:

- **Static content groups**. Finds an exact match between the URL stem and host name on the request and the URL stem and host name of the response.
- **Dynamic content groups**. Looks for objects that contains particular parameter-value pairs, arbitrary strings, or string patterns. Dynamic content groups are useful when caching data that is updated frequently (for example, a bug report or a stock quote).

Process overview: Serving a hit from a content group

1. A user enters search criteria for an item, such as a bug report, and clicks the Find button in an HTML form.
2. The browser issues one or more HTTP GET requests. These requests contain parameters (for example, the bug owner, bug ID, and so on).
3. When the NetScaler appliance receives the requests, it searches for a matching policy, and if it finds a caching policy that matches these requests, it directs the requests to a content group.
4. The content group looks for appropriate objects in the content group, usually based on criteria that you configure in a selector.
   For example, the content group can retrieve responses that match NameField=username and BugID=ID.
5. If it finds matching objects, the NetScaler appliance can serve them to the user's browser, where they are assembled into a complete response (for example, a bug report).

Example: Invalidating an object in a content group

1. A user modifies data (for example, the user modifies the bug report and clicks the Submit button).
2. The browser sends this data in the form of one or more HTTP requests. For example, it can send a bug report in the form of several HTTP POST requests that contain information about the bug owner and bug ID.
3. The NetScaler appliance matches the requests against invalidation policies. Typically, these policies are configured to detect the HTTP POST method.
4. If the request matches an invalidation policy, the NetScaler appliance searches the content group that is associated with this policy, and expires responses that match the configured criteria for invalidation.
   For example, an invalidation selector can find responses that match NameField=username and BugID=ID.
5. The next time the NetScaler appliance receives a GET request for these responses, it fetches refreshed versions from the origin server, caches the refreshed responses, and serves these responses to the user's browser, where they are assembled into a complete bug report.

Setting Up a Basic Content Group

Aug 07, 2014

By default, all cached data is stored in the default content group. You can configure additional content groups and specify these content groups in one or more policies.

You can configure content groups for static content, and you must configure content groups for dynamic content. You can modify the configuration of any content group, including the default group.

To set up a basic content group by using the command line interface

At the command prompt, type:

add cache contentgroup <name> (-hitSelector <hitSelectorName> -invalSelector <invalidationSelectorName> | -hitParams <hitParamName> -invalParams <invalidationParamName>) -type <type> [-relExpiry <sec> | -relExpiryMilliSec <msec>] [-heurExpiryParam <positiveInteger>]

Examples

```
> add cache contentgroup Products_Details –hitSelector product_selector –invalSelector id_selector
> add cache contentgroup bugrep -hitParams IssuePage RecordID Template TableId -invalParams RecordID -relExpiry 864000
```

To set up a basic content group by using the configuration utility

Navigate to Optimization > Integrated Caching > Content Groups, and create the content group.

Expiring or Flushing Cached Objects

Oct 28, 2013

If a response does not have an Expires header or a Cache-Control header with an expiration time (Max-Age or Smax-Age), you must expire objects in a content group by using one of the following methods:

- Configure content group expiration settings to determine whether and how long to keep the object.
- Configure an invalidation policy and action for the content group. For more information, see "Configuring Policies for Caching and Invalidation."
- Expire the content group or objects within it manually.

After a cached response expires, the NetScaler appliance refreshes it the next time the client issues a request for the response. By default, when the cache is full, the NetScaler appliance replaces the least recently used response first.

The following list describes methods for expiring cached responses using settings for a content group. Typically, these methods are specified as a percent or in seconds:

- **Manual**. Manually invalidate all responses in a content group or all responses in the cache.
- **Response-based**. Specific expiration intervals for positive and negative responses. Response-based expiry is considered only if the Last-Modified header is missing in the response.
- **Heuristic expiry**. For responses that have a Last-Modified header, heuristic expiry is a percentage of the time since the response was modified (calculated as current time minus the Last-Modified time, multiplied by the heuristic expiry value). For example, if a Last-Modified header indicates that a response was updated 2 hours ago, and the heuristic expiry setting is 10%, cached objects expire after 0.2 hours. This method assumes that frequently updated responses need to be expired more often.
- **Absolute or relative**. Specify an exact (absolute) time when the response expires every day, in HH:MM format, local time or GMT. Local time may not work in all time zones.
  Relative expiration specifies a number of seconds or milliseconds from the time a cache miss causes a trip to the origin server to the expiration of the response. If you specify relative expiration in milliseconds, enter a multiple of 10. This form of expiration works for all positive responses. Last-Modified, Expires, and Cache-Control headers in the response are ignored.

  Absolute and relative expiration override any expiration information in the response itself.

- **On download**. The option Expire After Complete Response Received expires a response as soon as it is downloaded. This is useful for frequently updated responses, for example, stock quotes. By default, this option is disabled.
  Enabling both Flash Cache and Expire After Complete Response Received accelerates the performance of dynamic applications. When you enable both options, the NetScaler appliance fetches only one response for a block of simultaneous requests.

  For more information, see "Queuing Requests to the Cache."

- **Pinned**. By default, when the cache is full the NetScaler appliance replaces the least recently used response first. The NetScaler appliance does not apply this behavior to content groups that are marked as pinned.

If you do not configure expiration settings for a content group, the following are additional options for expiring objects in the group:

- Configure a policy with an INVAL action that applies to the content group.

- Enter the names of content groups when configuring a policy that uses an INVAL action.

How Expiration Methods Are Applied

Expiration works differently for positive and negative responses. Positive and negative responses are described in the table, *Expiration of Positive and Negative Responses* mentioned below.

The following are rules of thumb for understanding the expiration method that is applied to a content group:

- You can control whether the NetScaler appliance evaluates response headers when deciding whether to expire an object.
- Absolute and relative expiration cause the NetScaler appliance to ignore the response headers (they override any expiration information in the response).
- Heuristic expiration settings and “Weak Positive” and “Weak Negative” expiration (labeled as **Default** values in the configuration utility) cause the NetScaler appliance to examine the response headers. These settings work together as follows:
  - The value in an Expires or Cache-Control header overrides these content group settings.
  - For positive responses that lack an Expires or Cache-Control header but have a Last-Modified header, the NetScaler appliance compares heuristic expiration settings with the header value.
  - For positive responses that lack an Expires, Cache-Control, or Last-Modified header, NetScaler appliance uses the “weak positive” value.
  - For negative responses that lack an Expires or Cache-Control header, NetScaler appliance uses the “weak negative” value.

For a list of expiration parameters see, "Configuring Periodic Expiration of a Content Group." The following table describes how these methods are applied.

Table 1. Expiration of Positive and Negative Responses

Response Type	Expiration Header Type	Content Group Setting	Period the Object Remains in the Cache
Positive	any header	Expire Content After (relExpiry) with no other settings	Use the value of the Expire Content After setting.
Positive	any header	Expire Content At (absExpiry) with no other settings	Subtract current date from the value of the Expire Content At setting.
Positive	any header	Expire Content After (relExpiry) and Expire content at (absExpiry)	Use the smaller of the two values for the content group settings. See the previous rows in this table.
Positive	Last-Modified (with any other headers)	Heuristic (heurExpiry Param) with any other setting	Subtract the Last-Modified date from the current date, multiply the result by the value of the heuristic expiry setting, and then divide by 100.

Response Type	Expiration Header Tags	Content Group Setting	Period the Object Remains in the Cache
Positive	Last-Modified (with any other headers)	Default (positive) (weakPosRel Expiry) and no other setting	Use the value of the Default (positive) expiry setting.
Positive	Expires or Cache-Control: Max-Age header is present Last-Modified header is absent	Heuristic (heurExpiry Param), Default (positive) (weakPosRel Expiry), or both	Subtract the current date from the Expires or the Cache-Control:Max-Age date.
Positive	no caching headers	Default (positive) (weakPosRel Expiry) and any other expiration setting.	Use the value of the Default (positive) setting.
Positive	no caching headers	Heuristic (heurExpiry Param) is present Default (positive) (weakPosRel Expiry) setting is absent	If the Last-Modified header is absent, the response is not cached or it is cached with an Already Expired status. If the Last-Modified header is present, use the heuristic expiry value.
Negative	Expires or Cache-Control:Max-Age	Expire Content After (relExpiry), Expire Content At (absExpiry), or both settings	Subtract the current date from the value of the Expires header, or use the value of the Cache-Control:Max-Age header.
Negative	Expires or Cache-Control headers are absent	Expire Content After (relExpiry), Expire Content At (absExpiry), or both settings	Response is not cached, or is cached with an Already Expired status.
Negative	Expires or Cache-Control:Max-Age	Any setting	Subtract the current date from the Expires or Cache-Control:Max-Age date.
Negative	Expires and Cache-Control:Max-Age headers are absent	Default (negative) (weakNegRel Expiry)	Use the value of the Default (negative) setting.
Negative	Expires and Cache-Control:Max-Age headers are absent	Any setting other than Default (negative) (weakNegRel Expiry)	Object is not cached or is cached with an Already Expired status.

Expiring a Content Group Manually

Aug 04, 2014

You can manually expire all of the entries in a content group.

To manually expire all responses in a content group by using the command line interface

At the command prompt, type:

expire cache contentGroup <name>

To manually expire all responses in a content group by using the configuration utility

Navigate to Optimization > Integrated Caching > Content Groups, select the content group, and click Invalidate to expire all the responses in a content group.

To manually expire all responses in the cache by using the configuration utility

Navigate to Optimization > Integrated Caching > Content Groups, and click Invalidate All to expire all the responses in cache.

Configuring Periodic Expiration of a Content Group

Aug 08, 2014

You can configure a content group so that it performs selective or full expiration of its entries. The expiration interval can be fixed or relative.

To configure content group expiration by using the command line interface

At the command prompt, type:

set cache contentgroup <name> (-relExpiry | -relExpiryMilliSec | -absExpiry | -absExpiryGMT | -heurExpiryParam | -weakPosRelExpiry | -weakNegRelExpiry | -expireAtLastBye) <expirationValue>

To configure content group expiration by using the configuration utility

Navigate to Optimization > Integrated Caching > Content Groups, select the content group, and specify expiry method.

Expiring Individual Responses

Expiring a response forces the NetScaler appliance to fetch a refreshed copy from the origin server. Responses that do not have validators, for example, ETag or Last-Modified headers, cannot be revalidated. As a result, flushing these responses has the same effect as expiring them.

To expire a cached response in a content group for static data, you can specify a URL that must match the stored URL. If the cached response is part of a parameterized content group, you must specify the group name as well as the exact URL stem. The host name and the port number must be the same as in the host HTTP request header of the cached response. If the port is not specified, port 80 is assumed.

To expire individual responses in a content group by using the command line interface

At the command prompt, type:

expire cache object -url <URL> -host <hostName> [-port <port>] [-groupName <contentGroupName>] [-httpMethod GET | POST]

To expire individual responses in a content group by using the command line interface (selector-based)

At the command prompt, type the following command:

expire cache object -locator <positiveInteger>

To expire a cached response by using the configuration utility

Navigate to Optimization > Integrated Caching > Cached Objects, select the cached response, and expire.

To expire a response by using the Lookup tool (selector-based)

Navigate to Optimization > Integrated Caching > Cached Objects, click Search and, set the search criteria to find the required cached response and expire.

Flushing Responses in a Content Group

You can remove, or flush, all responses in a content group, some responses in a group, or all responses in the cache. Flushing

a cached response frees up memory for new cached responses.

Note: To flush responses for more than one object at a time, use the configuration utility method. The command line interface does not offer this option.

To flush responses from a content group by using the command line interface

At the command prompt, type:

flush cache contentGroup <name> [-query <queryString> | [-selectorValue <selectorExpressionIDList> -host <hostName>]]

To flush responses from a content group by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups.
2. In details pane, flush the responses as follows:
   - To flush all responses in all content groups, click Invalidate All, and flush all the responses.
   - To flush responses in a particular content group, select the content group, click Invalidate, and flush all the responses.

Note: If this content group uses a selector, you can selectively flush responses by entering a string in the Selector value text box, entering a host name in the Host text box. Then click Flush and OK. The Selector value can be a query string of up to 2319 characters that is used for parameterized invalidation.
If the content group uses an invalidation parameter, you can selectively flush responses by entering a string in the Query field.

If the content group uses an invalidation parameter and Invalidate objects belonging to target host is configured, enter strings in the Query and Host fields.

To flush a cached response by using the command line interface

At the command prompt, type:

flush cache object -locator <positiveInteger> | -url <URL> -host <hostName> [-port <port>] [-groupName <contentGroupName>] [-httpMethod GET | POST]

To flush a cached response by using the configuration utility

Navigate to Optimization > Integrated Caching > Cached Objects, select the cached object, and flush.

Deleting a Content Group

You can remove a content group if it is not used by any policy that stores responses in the cache. If the content group is bound to a policy, you must first remove the policy. Removing the content group removes all responses stored in that group.

You cannot remove the Default, BASEFILE, or Deltajs group. The Default group stores cached responses that do not belong in any other content group.

To delete a content group by using the command line interface

At the command prompt, type:

rm cache contentgroup<name>

To delete a content group by using the configuration utility

Navigate to Optimization > Integrated Caching > Contetn Groups, select the content group, and delete.

Configuring Policies for Caching and Invalidation

Aug 30, 2013

Policies enable the integrated cache to determine whether to try to serve a response from the cache or the origin. The Citrix NetScaler appliance provides built-in policies for integrated caching, and you can configure additional policies. When you configure a policy, you associate it with an action. An action either caches the objects to which the policy applies or invalidates (expires) the objects. Typically, you based caching policies on information in GET and POST requests. You typically base invalidation policies on the presence of the POST method in requests, along with other information. You can use any information in a GET or POST request in a caching or an invalidation policy.

You can view some of the built-in policies in the integrated cache's Policies node in the configuration utility. The built-in policy names begin with an underscore (_).

Actions determine what the NetScaler appliance does when traffic matches a policy. The following actions are available:

- **Caching actions**. Policies that you associate with the CACHE action store responses in the cache and serve them from the cache.
- **Invalidation actions**. Policies that you associate with the INVAL action immediately expire cached responses and refresh them from the origin server. Note that for Web-based applications, invalidation policies often evaluate POST requests.
- **"Do not cache" actions**. Policies that you associate with a NOCACHE action never store objects in the cache.
- **"Provisionally cache" actions**. Policies that you associate with a MAYCACHE or MAYNOCACHE action depend on the outcome of additional policy evaluations.

Although the integrated cache does not store objects specified by the LOCK method, you can invalidate cached objects upon receipt of a LOCK request. For invalidation policies only, you can specify LOCK as a method by using the expression `http.req.method.eq("lock")`. Unlike policies for GET and POST requests, you must enclose the LOCK method in quotes because the NetScaler appliance recognizes this method name as a string only.

After you create a policy, you bind it to a particular point in the overall processing of requests and responses. Although you create a policy before binding it, you should understand how the bind points affect the order of processing before you create your policies.

The policies bound to a particular bind point constitute a policy bank. You can use goto expressions to modify the order of execution in a policy bank. You can also invoke policies in other policy banks. In addition, you can create labels and bind policies to them. Such a label is not associated with a processing point, but the policies bound to it can be invoked from other policy banks.

Actions to Associate with Integrated Caching Policies

Mar 16, 2012

The following table describes actions for integrated caching policies.

Table 1. Actions That You Can Associate with an Integrated Caching Policy

Action	Specifies
CACHE	Serves a response from the cache if the response has not expired. If the response must be fetched from the origin server, the NetScaler appliance caches the response before serving it. Even data that is updated and accessed frequently can be cached. For example, stock quotes are updated frequently, but they can be cached so that they can be served quickly to multiple users. If necessary, cached data can be refreshed immediately after it is downloaded. A CACHE action can be overridden by built-in policies.
NOCACHE	Always fetches the response from the origin server and marks the response as non-storable. You typically configure NOCACHE policies for data that is sensitive or personalized.
MAY_CACHE	Used in a request-time policy, this setting provisionally enables a response to be stored in a content group, pending evaluation of response-time policies. The following are possible: • If a matching response-time policy has a CACHE action but does not specify a content group, the response is stored in the Default group unless built-in policies override this policy. • If a matching response-time policy has a CACHE action and specifies the same content group as the one in the request-time policy, the response is stored in the named content group unless built-in policies override this policy. • If a matching response-time policy has a CACHE action but specifies a different content group from the one in the request-time policy, a NOCACHE action is applied. • If a matching response-time policy has a NOCACHE action, perform a NOCACHE action. • If there is no matching response-time policy, a CACHE action is applied, unless a built-in policy overrides this policy.
MAY_NOCACHE	For a request-time policy, this setting provisionally prevents caching the response. At response time, one of following actions is taken: • If no response-time policy matches the request, the final action is NOCACHE. • If a matching response-time policy contains a CACHE action, the final action is CACHE, unless built-in policies override this policy. • If a matching response-time policy contains a NOCACHE action, the final action is NOCACHE. • If a matching response-time policy has a CACHE action but does not specify a content group, the final action is to CACHE the response in the Default content group, unless built-in policies override this policy.

Action	Specifies
INVAL	Expires cached responses. Depending on how the policy and the content group are configured, all responses in one or more content groups are expired, or selected objects in the content group are expired. Note: You can specify INVAL actions in request-time policies only.

Bind Points for a Policy

Oct 28, 2013

You can bind the policy to one of the following bind points:

- **A global policy bank**. These are the request-time default, request-time override, response-time default, and response-time override policy banks, as described in "Order of Policy Evaluation."
- **A virtual server**. Policies that you bind to a virtual server are processed after the global override policies and before the global default policies, as described in "Order of Policy Evaluation." Note that when binding a policy to a virtual server, you bind it to either request-time or response-time processing.
- **An ad-hoc policy label**. A policy label is a name assigned to a policy bank. In addition to the global labels, the integrated cache has two built-in custom policy labels:
  - **_reqBuiltinDefaults**. This policy label, by default, is invoked from the request-time default policy bank.
  - **_resBuiltinDefaults**. This policy label, by default, is invoked from the response-time default policy bank.

  You can also define new policy labels. Policies bound to a user-defined policy label must be invoked from within a policy bank for one of the built-in bind points. For more information about creating a policy label, see "Configuring a Policy Label in the Integrated Cache." For more information about policy label invocation, see "Configuring a Policy Bank for Caching."

Important: You should bind a policy with an INVAL action to a request-time override or a response-time override bind point. To delete a policy, you must first unbind it.

Order of Policy Evaluation

For an advanced policy to take effect, you must ensure that the policy is invoked at some point during the NetScaler appliance's processing of traffic. To specify the invocation time, you associate the policy with a bind point. The following are the bind points, listed in order of evaluation:

- **Request-time override**. If a request matches a request-time override policy, by default request-time policy evaluation ends and the NetScaler appliance stores the action that is associated with the matching policy.
- **Request-time load balancing virtual server**. If policy evaluation cannot be completed after all the request-time override policies are evaluated, the NetScaler appliance processes request-time policies that are bound to load balancing virtual servers. If the request matches one of these policies, evaluation ends and the NetScaler appliance stores the action that is associated with the matching policy.
- **Request-time content switching virtual server**. Policies that are bound to this bind point are evaluated after request-time policies that are bound to load balancing virtual servers.
- **Request-time default**. If policy evaluation cannot be completed after all request-time, virtual server-specific policies are evaluated, the NetScaler appliance processes request-time default policies. If the request matches a request-time default policy, by default request-time policy evaluation ends and the NetScaler appliance stores the action that is associated with the matching policy.
- **Response-time override**. Similar to request-time override policy evaluation.
- **Response-time load balancing virtual server**. Similar to request-time virtual server policy evaluation.
- **Response-time content switching virtual server**. Similar to request-time virtual server policy evaluation.
- **Response-time default**. Similar to request-time default policy evaluation.

You can associate multiple policies with each bind point. To control the order of evaluation of the policies associated with the bind point you configure a priority level. In the absence of any other flow control information, policies are evaluated according to priority level, starting with the lowest numeric priority value.

After all integrated caching policies have been evaluated, if there are conflicting actions specified in request-time and response-time policies, the NetScaler appliance determines the final action as specified in the table, "Actions That You Can Associate with an Integrated Caching Policy."

Note: Request-time policies for POST data or cookie headers must be invoked during request-time override evaluation, because the built-in request-time policies in the integrated cache return a NOCACHE action for POST requests and a MAY_NOCACHE action for requests with cookies. Note that you would associate MAY_CACHE or MAY_NOCACHE actions with a request-time policy that points to a parameterized content group. The response time policy determines whether the transaction is stored in the cache.

Configuring a Policy in the Integrated Cache

Aug 07, 2014

You configure new policies to handle data that the built-in policies cannot process. You configure separate policies for caching, preventing caching from occurring, and for invalidating cached data. Following are the main components of a policy for integrated caching:

- Rule: A logical expression that evaluates an HTTP request or response.
- Action: You associate a policy with an action to determine what to do with a request or response that matches the policy rule.
- Content groups: You associate the policy with one or more content groups to identify where the action is to be performed.

To configure a policy for caching by using the command line interface

At the command prompt, type:

add cache policy <policyName> -rule <expression> -action CACHE | MAY_CACHE | NOCACHE | MAY_NOCACHE [-storeInGroup <contentGroupName>] [-undefAction NOCACHE | RESET]

Examples

```
> add cache policy image_cache -rule "http.req.url.contains(\"jpg\") || http.req.url.contains(\"jpeg\")" -action CACHE -storeingroup myImages_group -undefaction NOCACHE
> add cache policy bugReportPolicy -rule "http.req.url.query.contains(\"IssuePage\")" -action CACHE -storeInGroup bugReportGroup
> add cache policy my_form_policy -rule "http.req.header(\"Host\")contains(\"my.company.com\") && http.req.method.eq(\"GET\") && http.req.url.query.contains(\"v=7\")" -
> add cache policy viewproducts_policy –rule "http.req.url.contains(\"viewproducts.aspx\")" –action CACHE -storeInGroup Product_Details
```

To configure a policy for invalidation by using the command line interface

At the command prompt, type:

add cache policy <policyName> -rule <expression> -action INVAL [-invalObjects "<contentGroupName1>[,<selectorName1>"]. . .]] | [-invalGroup <contentGroupName1>[, <contentGroupName2>. . .]] [-undefAction NOCACHE | RESET]

Examples

```
> add cache policy invalidation_events_policy -rule "http.req.header(\"Host\")contains(\"my.company.com\") && http.req.method.eq(\"GET\") && http.req.url.query.contains
> add cache policy inval_all -rule "http.req.method.eq(\"POST\") && http.req.url.contains(\"jpeg\")" -action INVAL -invalGroups myImages_group myApps_group PDF_group
> add cache policy bugReportInvalidationPolicy -rule "http.req.url.query.contains(\"TransitionForm\")" -action INVAL -invalObjects bugReport
> add cache policy editproducts_policy –rule "http.req.url.contains(\"editproducts.aspx\")" –action INVAL -invalObjects "Product_Details,batchnum_sel" "Products_In_Depo
```

To configure a policy for caching or invalidation by using the configuration utility

Navigate to Optimization > Integrated Caching > Policies, and create the new policy.

Globally Binding an Integrated Caching Policy

Aug 08, 2014

When you globally bind a policy, it is available to all virtual servers on the NetScaler appliance.

To bind an integrated caching policy globally by using the command line interface

At the command prompt, type:

bind cache global <policy> -priority <positiveInteger> [-type REQ_OVERRIDE | REQ_DEFAULT | RES_OVERRIDE | RES_DEFAULT] [-gotoPriorityExpression <expression>] [-invoke <labelType> <labelName>]

Example

```
> bind cache global myCachePolicy -priority 100 -type req_default
```

Note that the type argument is optional for globally bound policies, to maintain backward compatibility with policies that you defined using earlier versions of the NetScaler appliance. If you omit the type, the policy is bound to REQ_DEFAULT or RES_DEFAULT, depending on whether the policy rule is a response-time or a request-time expression. If the rule contains both request time and response time parameters, it is bound to RES_DEFAULT. Following is an example of a binding that omits the type.

```
> bind cache global myCache Policy 200
```

To bind an integrated caching policy globally by using the configuration utility

1. Navigate to Optimization > Integrated Caching.
2. In the details pane, click Cache policy manager.
3. In the Cache Policy Manager dialog box, select a Request or Response bind point, and then select a second level of binding of either Override Global or Default Global. A list of policies appears. These are policies that are bound to this bind point.
4. Click Insert Policy and do one of the following:
   - To configure a new policy, click New Policy and configure the new policy as described in "Configuring a Policy in the Integrated Cache."
   - To bind an existing policy, click the name of the policy.
5. Drag and drop the policy to the position in the policy bank where you want it to be evaluated, or manually enter a priority level, as a positive integer, for this entry in the Priority field.
6. Optionally, to configure a Goto expression as described in the "Policies and expressions", double-click the field in the Goto Expression column, and enter valid priority level, the keywords NEXT or END, or an advanced expression. See "Entries to Control Evaluation Flow in a Policy Bank" for details.
7. Optionally, to invoke an external policy bank, click the field in the Invoke Type column, and select the type of policy bank that you are adding (a global label or a virtual server bank). In the Invoke Name field, enter the label or virtual server name. See "Entries to Control Evaluation Flow in a Policy Bank" for details.
8. Click Apply Changes.

Binding an Integrated Caching Policy to a Virtual Server

Aug 08, 2014

When you bind a policy to a virtual server, it is available only to requests and responses that match the policy and that flow through the relevant virtual server.

When using the configuration utility, you can bind the policy using the configuration dialog box for the virtual server. This enables you to view all of the policies from all NetScaler modules that are bound to this virtual server. You can also use the Policy Manager configuration dialog for the integrated cache. This enables you to view only the integrated caching policies that are bound to the virtual server.

To bind an integrated caching policy to a virtual server by using the command line interface

At the command prompt, type:

- bind lb vserver <name>@ -policyName <policyName> -priority <positiveInteger> -type (REQUEST | RESPONSE)
- bind cs vserver <name>@ -policyName <policyName> -priority <positiveInteger> -type (REQUEST | RESPONSE)

To bind an integrated caching policy to a virtual server by using the configuration utility (virtual server method)

- CS Virtual Server - Navigate toTraffic Management > Content Switching > Virtual Servers, select the virtual server, and bind relevant cache policies.
- LB Virtual Server - Navigate toTraffic Management > Load Balancing > Virtual Servers, select the virtual server, and bind relevant cache policies.

To bind an integrated caching policy to a virtual server by using the configuration utility (Policy Manager method)

1. Navigate to Optimization > Integrated Caching.
2. In the details pane, click Cache policy manager.
3. In the Cache Policy Manager dialog box, select a Request or Response bind point, select a second level of binding of either LB Virtual Server or CS Virtual Server, and then select the name of a virtual server. A list of policies appears. These are integrated caching policies that are bound to this virtual server.
4. Click Insert Policy and do one of the following:
   - To configure a new policy, click New Policy and configure the new policy as described in "Configuring a Policy in the Integrated Cache."
   - To bind an existing policy, click the name of the policy.
5. Drag and drop the policy to the position in the policy bank where you want it to be evaluated, or manually enter a priority level, as a positive integer, for this entry in the Priority field.
6. Optionally, configure a Goto expression as described in "Configuring a Policy Bank for Caching."
7. Optionally, to invoke an external policy bank, click the field in the Invoke Type column, and select the type of policy bank that you are adding (a global label or a virtual server bank). In the Invoke Name field, enter the label or virtual server name. See "Entries to Control Evaluation Flow in a Policy Bank."
8. Click Apply Changes.

Example: Caching Compressed and Uncompressed Versions of a File

Mar 16, 2012

By default, a client that can handle compression can be served uncompressed responses or compressed responses in gzip, deflate, compress, and pack200-gzip format. If the client handles compression, an Accept-Encoding:compression format header is sent in the request. The compression type accepted by the client must match the compression type of the cached object. For example, a cached .gzip file cannot be served in response to a request with an Accept-Encoding:deflate header.

A client that cannot handle compression is served a cache miss if the cached response is compressed.

For dynamic caching, you need to configure two content groups, one for compressed data and one for uncompressed versions of the same data. The following is an example of configuring the selectors, content groups, and policies for serving uncompressed files from the cache to clients that cannot handle compression, and serving compressed versions of the same files to client that can handle compression.

```
add cache selector uncompressed_response_selector http.req.url "http.req.header(\"Host\")"
add cache contentGroup uncompressed_group -hitSelector uncompressed_responst_selector -invalSelector uncomp_resp_sel
add cache policy cache_uncompressed -rule "HTTP.REQ.URL.CONTAINS(\"xyz\") && !HTTP.REQ.HEADER(\"Accept-Encoding\").EXISTS" -action CACHE -storeInGroup uncomp
bind cache global cache_uncompressed -priority 100 -gotoPriorityExpression END -type REQ_OVERRIDE
add cache selector compressed_response_selector HTTP.REQ.URL "HTTP.REQ.HEADER(\"Host\")" "HTTP.REQ.HEADER(\"Accept-Encoding\")"
add cache contentGroup compressed_group -hitSelector compressed_response_selector
add cache policy cache_compressed -rule "HTTP.REQ.URL.CONTAINS(\"xyz\") && HTTP.REQ.HEADER(\"Accept-Encoding\").EXISTS" -action CACHE -storeInGroup compresse
bind cache global cache_compressed -priority 200 -gotoPriorityExpression END -type REQ_OVERRIDE
```

Configuring a Policy Bank for Caching

Aug 08, 2014

All of the policies that are associated with a particular bind point are collectively known as a policy bank. In addition to configuring priority levels for policies in a bank, you can modify the order of evaluation order in a bank by configuring Goto expressions. You can further modify the evaluation order by invoking an external policy bank from within the current policy bank. You can also configure new policy banks, to which you assign your own labels. Because such policy banks are not bound to any point in the processing cycle, they can be invoked only from within other policy banks. For convenience, policy banks whose labels do not correspond to a built-in bind point are called policy labels.

In addition to controlling order of policy evaluation by binding the policy and assigning a priority level, as described in "Binding Policies That Use the Default Syntax", you can establish the flow within a bank of policies by configuring a Goto expression. A Goto expression overrides the flow that is determined by the priority levels. You can also control the evaluation flow by invoking an external policy bank after evaluating an entry in the current bank. Evaluation always returns to the current bank after evaluation has completed for the external bank.

The following table summarizes the entries to control evaluation in a policy bank.

Table 1. Entries to Control Evaluation Flow in a Policy Bank

Attribute	Specifies
Name	The name of a policy, or, to invoke another policy bank without evaluating the policy, the keyword NOPOLICY. You can specify NOPOLICY more than once in a policy bank, but you can specify a named policy only once.
Priority	An integer. The lower the integer, the higher the priority.
Goto Expression	Determines the next policy or policy bank to evaluate. You can provide one of the following values: • **NEXT**: Go to the policy with the next higher priority. • **END**: Stop evaluation. • **USE_INVOCATION_RESULT**: Applicable if this entry invokes another policy bank. If the final Goto in the invoked bank has a value of END, evaluation stops. If the final Goto is anything other than END, the current policy bank performs a NEXT. • **Positive number**: Priority number of the next policy to be evaluated. • **Numeric expression**: Expression that produces the priority number of the next policy to be evaluated. The Goto can only proceed forward in a policy bank. Omitting the Goto expression is the same as specifying END.
Invocation Type	Designates a policy bank type. The value can be one of the following: • **Request Vserver**: Invokes request-time policies that are associated with a virtual server.

Attribute	Specifies
	• **Response Vserver**: Invokes response-time policies that are associated with a virtual server. • **Policy label**: Invokes another policy bank, as identified by the policy label for the bank.
Invocation Name	Name of a virtual server or a policy label, depending on the value that you specified for the Invocation Type.

The integrated cache has two built-in policy labels, and you can configure additional policy labels:

- **_reqBuiltInDefaults**: This policy label is invoked from the request-time default bind point.
- **_resBuiltInDefaults**: This policy label is invoked from the response-time default bind point.

Note: For information about creating policy labels, see "Configuring a Policy Label in the Integrated Cache."

To invoke a policy label in a caching policy bank by using the command line interface

At the command prompt, type:

bind cache policylabel <labelName> -policname<policyName> -priority<priority> [-gotoPriorityExpression <gotopriorityExpression>] [-invoke <labelType> <labelName>]

To invoke a policy label in a caching policy bank by using the configuration utility

1. Navigate to Optimization > Integrated Caching, click Cache policy manager, and specify the relevant bind point (Override Global or Default Global) and connection type to view the list of policies bound to this bind point.
2. If you want to invoke a policy label without evaluating a policy, click NOPOLICY.
   Note: To invoke an external policy bank, click the field in the Invoke Type column, and select the type of policy bank that you want to invoke at this point in the policy bank. This can be a global label or a virtual server bank. In the Invoke Name field, enter the label or virtual server name. See "Entries to Control Evaluation Flow in a Policy Bank" for details.

To invoke a caching policy label in a virtual server policy bank by using the command line interface

At the command prompt, type:

- bind lb vserver <name>@ -policyName <policyName>|<NOPOLICY-CACHE> -priority <positiveInteger> -gotoPriorityExpression <expression> -type REQUEST|RESPONSE -invoke <labelType> <labelName>
- bind cs vserver <name> -policyName <policyName>|<NOPOLICY-CACHE> -priority <positiveInteger> -gotoPriorityExpression <expression> -type REQUEST|RESPONSE -invoke <labelType> <labelName>

For more information, see "Entries to Control Evaluation Flow in a Policy Bank."

To invoke a caching policy label in a virtual server policy bank by using the configuration utility

1. Navigate to Traffic Management > Load Balancing/Contetn Switching > Virtual Servers, select the virtual server, and click Policies.
2. If you are configuring an existing entry in this bank, skip this step. If you are adding a new policy to this policy bank, or you want to use the “dummy” NOPOLICY entry, click Add, and do one of the following:
   - To configure a new policy, click Cache and configure the new policy as described in "Configuring a Policy in the Integrated Cache."
   - To invoke a policy bank without processing a policy a rule, select the NOPOLICY-CACHE option.

   Note: To invoke an external policy bank, click the field in the Invoke Type column, and select the type of policy bank that you want to invoke at this point in the policy bank. This can be a global label or a virtual server bank. In the Invoke Name

field, enter the label or virtual server name. See "Entries to Control Evaluation Flow in a Policy Bank" for details.

Configuring a Policy Label in the Integrated Cache

Aug 08, 2014

In addition to configuring policies in a policy bank for one of the built-in bind points or a virtual server, you can create caching policy labels and configure banks of policies for these new labels.

A policy label for the integrated cache can be invoked only from one of the bind points that you can view in the Policy Manager in the **Integrated Caching** details pane (request override, request default, response override, or response default) or the built-in policy labels _reqBuiltinDefaults and _resBuiltinDefaults. You can invoke a policy label any number of times unlike a policy, which can only be invoked once.

The configuration utility provides an option to rename a policy label. Renaming a policy label does not affect the process of evaluation of the policies bound to the label.

Note: You can use the NOPOLICY “dummy” policy to invoke any policy label from another policy bank. The NOPOLICY entry is a placeholder that does not process a rule.

To configure a policy label for caching by using the command line interface

At the command prompt, type the following command to create a policy label and verify the configuration:

- add cache policylabel <labelName> -evaluates (REQ | RES)
- show cache policylabel <labelName>

Invoke this policy label from a policy bank. For more information, see "Configuring a Policy Bank for Caching."

To configure a policy label for caching by using the configuration utility

Navigate to Optimization > Integrated Caching > Policy Labels, add a policy label, and bind the cached policies.
Note: To ensure that the NetScaler ADC processes the policy label at the right time, configure an invocation of this label in one of the policy banks that are associated with the built-in bind points

To rename a policy label by using the configuration utility

Navigate to Optimization > Integrated Caching > Policy Labels, select the policy label, and rename.

Unbinding and Deleting an Integrated Caching Policy and Policy Label

Aug 08, 2014

You can unbind a policy from a policy bank, and you can delete it. To delete the policy, you must first unbind it. You can also remove a policy label invocation and delete a policy label. To delete the policy label, you must first remove any invocations that you have configured for the label.

You cannot unbind or delete the labels for the built-in bind points (request default, request override, response default, and response override).

To unbind a global caching policy by using the command line interface

At the command prompt, type:

unbind cache global <policy>

To unbind a virtual server-specific caching policy by using the command line interface

At the command prompt, type:

(unbind lb vserver | unbind cs vserver) <vserverName> -policyName <policyName> -type (REQUEST | RESPONSE)

To delete a caching policy by using the command line interface

At the command prompt, type:

rm cache policy <policyName>

To unbind a caching policy by using the configuration utility

Navigate to Optimization > Integrated Caching, click Cache Policy Manager, and unbind policies by specifying the relevant bind point and connection type (Request/Response).

To delete a policy label invocation by using the configuration utility

1. Navigate to Optimization > Integrated Caching, click Cache policy manager, and specify the relevant bing point (LB virtual server or CS virtual server) and connection type to view the list of cache policies bound to this virtual server.
2. In the policy Invoke column, clear the entry.

Caching Support for Database Protocols

Sep 02, 2013

The integrated cache monitors database requests that flow through the Citrix® NetScaler® appliance and caches them as determined by the cache policies. Users have to configure the cache polices for MYSQL and MSSQL protocols, because the NetScaler does not provide any default policies for these protocols. When configuring the protocols, remember that request based policies currently support CACHE and INVAL actions, while response based policies currently support only NOCACHE action. After configuring the policies, bind them to virtual servers. MYSQL and MSSQL policies, both request and response, can be bound only to virtual servers

Before creating a cache policy, create a cache content group of type MYSQL or MSSQL. When you create a MYSQL or MSSQL cache content group, associate at least one hit selector with it. See "Setting Up a Basic Content Group" for setting up cache content groups.

The following example illustrates the procedure for configuring and verifying cache support for SQL protocols.

```
> enable feature IC
> set cache parameter -memlimit 100
> add cache selector sel1 mssql.req.query.text

> add cache contentgroup cg1  -type "MSSQL" -hitselector "sel1" -invalselector "inval_sel" -relExpiry "500" -maxResSize
 "100"
> add cache policy cp1  -rule "mssql.req.query.command.contains(\"select\")" -action "CACHE" -storeInGroup "cg1"
> add cache policy cp2  -invalObjects "cg1" -rule "mssql.req.query.text.contains(\"insert\")" -action "INVAL"
> add db user user1  -password "Pass1"
> add service svc_sql_1 10.102.147.70 mssql 64834 -healthMonitor "NO" -downstateflush "ENABLED"
> add lb vserver lb_mssql1 mssql 10.102.147.77 1433  -lbmethod "roundrobin"
> bind lb vserver lb_mssql1 svc_sql_1
> bind lb vserver lb_mssql1 -policyName cp1  -type "REQUEST" -priority "2"
> bind lb vserver lb_mssql1 -policyName cp2  -type "REQUEST" -priority "1"

> show cache selector sel1
    Name:sel1
    Expressions:
    1)mssql.req.query.text
> show cache policy cp1
    Name:cp1
    Rule:mssql.req.query.command.contains("select")
    CacheAction:CACHE
    Stored in group: cg1
    UndefAction:Use Global
    Hits:2
    Undef Hits:0
    Policy is bound to following entities
    1) Bound to:
      REQ VSERVER lb_mssql1
      Priority:2
      GotoPriorityExpression: END
```

Note: The methods for reducing flash crowds, as explained in "Reducing Flash Crowds", are not supported for MYSQL and MSSQL protocols.

Configuring Expressions for Caching Policies and Selectors

Oct 28, 2013

A request-time expression examines data in request-time transaction, and a response-time expression examines data in a response-time transaction. In a policy for caching, if an expression matches data in a request or response, the Citrix NetScaler appliance takes the action associated with the policy. In a selector, request-time expressions are used to find matching responses that are stored in a content group.

Before configuring policies and selectors for the integrated cache, you need to know, at minimum, the host names, paths, and IP addresses that appear in HTTP request and response URLs. And you probably need to know the format of entire HTTP requests and responses. Programs such as Live HTTP Headers (http://livehttpheaders.mozdev.org/) or HTTPFox (https://addons.mozilla.org/en-US/firefox/addon/6647) can help you investigate the structure of the HTTP data that your organization works with.

Following is an example of an HTTP GET request for a stock quote program:

```
GET /quote.dll?page=dynamic&mode=data&mode=stock&symbol=CTXS&page=multi&selected=CTXS&random=0.00792039478975548 HTTP/1.1
Host: quotes.mystockquotes.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9) Gecko/2008052906 Firefox/3.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate,compress,pack200-gzip
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Referer: http://quotes.mystockquotes.com/quote.dll?mode=stock&symbol=CTXS&page=multi&selected=CTXS
Cookie: __qca=1210021679-72161677-10297606
```

When configuring an expression, note the following limitations:

Table 1. Restrictions on Request-Time and Response-Time Expressions

Expression Type	Restrictions
Request	Do not configure request-time expressions in a policy with a CACHE or NOCACHE action. Use MAY_CACHE or MAY_NOCACHE instead.
Response	Configure response-time expressions in caching policies only. • Selectors can use only request-time expressions. • Do not configure response-time expressions in a policy with an INVAL action. Do not configure response-time expressions in a policy with a CACHE action and a parameterized content group. Use the MAY_CACHE action.

Note: For a comprehensive discussion of advanced expressions, see "Policies and Expression."

Expression Syntax

Following are basic components of the syntax:

- Separate keywords with periods (.), as follows:
  `http.req.url`
- Enclose string values in parentheses and quotes, as follows:
  `http.req.url.query.contains("this")`
- When configuring an expression from the command line, you must escape internal quote marks (the quotes that delimit values in the expression, as opposed to the quotes that delimit the expression). One method is to use a slash, as followings:
  `\"abc\"`

Selector expressions are evaluated in order of appearance, and multiple expressions in a selector definition are joined by a logical AND. Unlike selector expressions, you can specify Boolean operators and modify the precedence in an advanced expression for a policy rule.

Configuring an Expression in a Caching Policy or a Selector

Updated: 2014-08-04

Note that on the command line, the syntax for a policy expression is somewhat different from a selector expression. For a comprehensive discussion of advanced expressions, see "Policies and Expressions."

To configure a policy expression by using the command line interface

1. Start the policy definition as described in "Globally Binding an Integrated Caching Policy."
2. To configure the policy rule, delimit the entire rule in quotes, and delimit string values within the rule in escaped quotes.

   The following is an example:

   ```
   "http.req.url.contains(\"jpg\")"
   ```
3. To add Boolean values, insert &&, ||, or ! operators.

   The following are examples:

   ```
   "http.req.url.contains(\"jpg\") || http.req.url.contains(\"jpeg\")"
   "http.req.url.query.contains(\"IssuePage\")"
   "http.req.header(\"Host\")contains(\"my.company.com\") && http.req.method.eq(\"GET\") && http.req.url.query.contains(\"v=7\")"
   ```
4. To configure an order of evaluation for the constituent parts of a compound

   ```
   "http.req.url.contains(\"jpg\") || (http.req.url.contains(\"jpeg\") && http.req.method.eq(\"GET\"))"
   ```

To configure a selector expression by using the command line interface

1. Start the selector definition as described in "About Content Groups."
2. To configure the selector expression, delimit the entire rule in quotes, and delimit string values within the rule in escaped quotes.

   The following is an example:

   ```
   "http.req.url.contains(\"jpg\")"
   ```
3. You cannot add Boolean values, insert &&, ||, or ! operators. Enter each expression element delimited in quotes. Multiple expressions in the definition are treated as a compound expression joined by logical ANDs.

   The following are examples:

   ```
   "http.req.url.query.value(\"ProductId\")" "http.req.url.query.value(\"BatchNum\")" "http.req.url.query.value(\"depotLocation\")"
   ```

To configure a policy or selector expression by using the configuration utility

1. Start the policy or selector definition as described in "To configure a policy for caching or invalidation by using the configuration utility" or "To configure a selector by using the configuration utility."
2. In the Expression field, you can either manually type the default syntax by clicking Switch to Classic Syntax or create new expression using Expression Editor.
3. To insert an operator between two parts of a compound expression, click the Operators button and select the operator type. The following is an example of a configured expression with a Boolean OR (signaled by double vertical bars, ||):
4. Click Frequently Used Expressions drop-down to insert the commonly used expressions.
5. To test the expression, click the Evaluate. In the Expression Evaluator dialog box, select the Flow Type that matches the expression. In the data field, paste the HTTP request or response that you hope to parse using the expression, and click Evaluate.

Displaying Cached Objects and Cache Statistics

May 19, 2015

You can view particular cached objects, and you can view summary statistics on cache hits, misses, and memory usage. The statistics provide insight on the amount of data that is being served from the cache, what items are responsible for the largest performance benefit, and what you can tune to improve cache performance.

This section includes the following details:

- Viewing Cached Objects
- Finding Particular Cached Responses
- Viewing Cache Statistics

Viewing Cached Objects

Updated: 2014-08-04

After enabling caching, you can view details for cached objects. For example, you can view the following items:

- Response sizes and header sizes
- Status codes
- Content groups
- ETag, Last-Modified, and Cache-Control headers
- Request URLs
- Hit parameters
- Destination IP addresses
- Request and response times

To view a list of cached objects by using the command line interface

At the command prompt, type:

show cache object

Table 1. Properties of Cached Objects

Properties	Specifies
Response size (bytes)	The size of the response header and body.
Response header size (bytes)	The size of the header portion of the response.
Response status code	The status code sent with the response.

Properties	Specifies
ETag	The ETag header inserted in the response. Typically, this header indicates whether the response has changed recently.
Last-Modified	The Last-Modified header inserted in the response. This header indicates the date that the response was last changed.
Cache-Control	The Cache-Control header inserted in the response.
Date	The Date header that indicates when the response was sent.
Contentgroup	The content group where the response is stored.
Complex match	If this object was cached on the basis of parameterized values, this field value is YES.
Host	The host specified in the URL that requested this response.
Host port	The listen port for the host specified in the URL that requested this response.
URL	The URL issued for the stored response.
Destination IP	The IP address of the server from which this response was fetched.
Destination port	The listen port for the destination server.
Hit parameters	If the content group that stores the response uses hit parameters, they are listed in this field.
Hit selector	If this content group uses a hit selector, it is listed in this field.
Inval selector	If this content group uses an invalidation selector, it is listed in this field.
Selector Expressions	If this content group uses a selector, this field displays the expression that defines the selection rule.
Request time	The time in milliseconds since the request was issued.

Response time	Specifies in milliseconds since the cache started to receive the response.
Age	Amount of time the object has been in the cache.
Expiry	Amount of time after which the object is marked as expired.
Flushed	Whether the response has been flushed after expiry.
Prefetch	If Prefetch has been configured for this content group, the amount of time before expiry during which the object is fetched from the origin. Prefetch does not apply to negative objects (for example, 404 "object not found" responses).
Current readers	Approximately the current number of hits being served. When a response with a Content-Length header object is being downloaded, the current misses and the current readers values are each typically 1. When a chunked response object is being downloaded, the current misses value is typically 1, but the current readers value is typically 0, because the chunked response that is served to the client does not come from the integrated caching buffers.
Current misses	The current number of requests that resulted in a cache miss and fetching from the origin server. This value is typically 0 or 1. If Poll Every Time is enabled for a content group, the count can be greater than 1.
Hits	The number of cache hits for this object.
Misses	The number of cache misses for this object.
Compression format	The type of compression applied to this object. Compression formats include gzip, deflate, compress, and pack200-gzip.
HTTP version in response	The version of HTTP that was used to send the response.
Weak etag present in response	Strong etag headers change if the bits of an entity change. Strong headers are based on the octet values of an object. Weak etag headers change if the meaning of an entity changes. Weak etag values are based on semantic identity. Weak etags values start with a "W."
Negative marker cell	A marker object is cacheable, but it does not yet meet all the criteria for being cached. For example, the object may exceed the maximum response size for the content group. A marker cell is created for objects of this type. The next time a user sends a request for this object, a cache miss is served.

Properties	Specifies
Reason marker created	The reason a marker cell was created (for example, “Waiting for minhit,” “Content-length response data is not in group size limit”).
Auto poll every time	If the integrated cache receives an already expired 200 OK response with validators (either the Last-Modified or the ETag response headers) it stores the response and marks it as Auto-PET (automatically poll every time).
NetScaler Etag inserted in response	A variation of the ETag header generated by the NetScaler appliance. A value of YES appears if the NetScaler inserts an Etag in the response.
Full response present in cache	Indicates whether this is a complete response.
Destination IP verified by DNS	Indicates whether DNS resolution was performed when storing the object.
Object stored through a cache forward proxy	Indicates whether this response was stored due to a forward proxy that is configured in the integrated cache.
Object is a Delta basefile	A response that is delta-compressed.
Waiting for minhits	Indicates whether this content group requires a minimum number of origin server hits before caching a response.
Minhit count	If this content group requires a minimum number of origin server hits before caching an object, this field displays a count of the number of hits received so far.
HTTP Request Method	The method, GET or POST, used in the request that obtained this object.
Stored by policy	The name of the caching policy that caused this object to be stored. A value of NOT AVAILABLE indicates that the policy has been deactivated or deleted. A value of NONE indicates that the object did not match a visible policy, but was stored according to internal criteria for caching.

Properties	Specifies
Application firewall metadata exists	This parameter is used when the application firewall and the integrated cache are both enabled. The application firewall analyzes the contents of a response page, stores its metadata (for example, URLs and forms contained in page), and exports the metadata with the response to the cache. The cache stores the page and the metadata, and when the cache serves the page, it sends the metadata back to the request's session.
HTTP callout object, name, type, response	These cells indicate whether this data was stored as a result of an HTTP Callout expression, and provide information about various aspects of the callout and the corresponding response. For more information about HTTP callouts, see "HTTP Callouts".

To view cached objects by using the configuration utility

To view cached objects by using the configuration utility

Navigate to Optimization > Integrated Caching > Cache Objects. You can view all the cached objects and sort them accordingly as per your requirement.

Finding Particular Cached Responses

Updated: 2014-08-08

You can find individual items in the cache based on search criteria. There are different methods for finding cached items, depending on whether the content group that contains the data uses hit and invalidation selectors, as follows:

- If the content group uses selectors, you can only conduct the search using the Locator ID for the cached item.
- If the content group does not use selectors, you conduct the search using criteria such as URL, host, content group name, and so on.

When searching for a cached response, you can locate some items by URL and host. If the response is in a content group that uses a selector, you can find it only by using a Locator number (for example, 0x00000000ad7af00000050). To save a Locator number for later use, right-click the entry and select **Copy**. For more information about selectors, see "Configuring Selectors and Basic Content Groups."

To display cached responses in content groups that do not have a selector by using the command line interface

At the command prompt, type:

show cache object [-locator <positiveInteger>] | [(-url <URL> (-host <hostName> [-port <port>] [-groupName <contentGroupName>] [-httpMethod GET | POST])) | [-httpStatus<positive integer>] | -group <contentGroupName> | -ignoreMarkerObjects (ON | OFF) | -includeNotReadyObjects (ON | OFF)]

To display cached responses in content groups that have a selector by using the command line interface

At the command prompt, type:

show cache object -locator <locatorString> MarkerObjects (ON | OFF) | -includeNotReadyObjects (ON | OFF) | [-

httpStatus<positive integer>]

To display cached responses in content groups that do not have a selector by using the configuration utility

Navigate to Optimization > Integrated Caching > Cache Objects, click Search, and set the search criteria to view the required cached response.
If you have not yet configured any content groups, all of the objects are in the Default group.

To display cached responses in content groups that have a selector by using the configuration utility

Navigate to Optimization > Integrated Caching > Cache Objects, click Search, and set the selector search criteria to view the required cached response.

Viewing Cache Statistics

Updated: 2013-10-28

The following table summarizes the detailed cache statistics that you can view.

Table 2. Integrated Cache Statistics

Counter	Specifies
Hits	Responses that are found in and served from the integrated cache. Includes static objects such as image files, pages with status codes 200, 203, 300, 301, 302, 304, 307, 403, 404, 410, and responses that match a user-defined policy with a CACHE action..
Misses	Intercepted HTTP requests where the response was ultimately fetched from origin server.
Requests	Total cache hits plus total cache misses.
Non-304 hits	If the user requests an item more than once, and the item in the cache is unchanged since the last time the NetScaler appliance served it, the NetScaler appliance serves a 304 response instead of the cached object. This statistic indicates how many items the NetScaler appliance served from the cache, excluding 304 responses.
304 hits	Number of 304 (object not modified) responses the NetScaler appliance served from the cache.
304 hit ratio (%)	Percentage of 304 responses that the NetScaler appliance served, relative to other responses.
Hit ratio (%)	Percentage of responses that the NetScaler appliance served from the cache (cache hits) relative to

Counter	Specifies
	responses that could not be served from the cache.
Origin bandwidth saved (%)	An estimate of the processing capacity that the NetScaler appliance saved on the origin server due to serving responses from the cache.
Bytes served by the NetScaler	Total number of bytes that the NetScaler appliance served from the origin server and the cache.
Bytes served by cache	Total number of bytes that the NetScaler appliance served from the cache.
Byte hit ratio(%)	Percentage of data that the NetScaler appliance served from the cache, relative to all of the data in all served responses.
Compressed bytes from cache	Amount of data, in bytes, that the NetScaler appliance served in compressed form.
Storable misses	If the NetScaler appliance does not find a requested object in the cache, it fetches the object from the origin server. This is known as a cache miss. A storable cache miss can be stored in the cache.
Non-storable misses	A non-storable cache miss cannot be stored in the cache.
Misses	All cache misses.
Revalidations	Max-Age setting in a Cache-Control header determines, in number of seconds, when an intervening cache must revalidate the content with the integrated cache before serving it to the user. For more information, see "Inserting a Cache-Control Header."
Successful revalidations	Number of re-validations that have been performed. For more information, see "Inserting a Cache-Control Header."
Conversions to conditional req	A user-agent request for a cached PET object is always converted to a conditional request and sent to the origin server. For more information, see "Polling the Origin Server Every Time a Request Is Received."

Counter	Specifies
Storable miss ratio (%)	Storable cache misses as a percentage of non-storable cache misses.
Successful reval ratio (%)	Successful revalidations as a percentage of all revalidation attempts. For more information, see "Inserting a Cache-Control Header."
Expire at last byte	Number of times that the cache expired content immediately after receiving the last body byte. Only applicable to positive responses, as described in the table "Cache Hits and Misses." For more information, see "Example of Performance Optimization."
Flashcache misses	If you enable Flash Cache, the cache allows only one request to reach the server, eliminating flash crowds. This statistic indicates the number of Flash Cache requests that were cache misses. For more information, "Queuing Requests to the Cache."
Flashcache hits	Number of Flash Cache requests that were cache hits. For more information, see "Queuing Requests to the Cache."
Parameterized inval requests	Requests that match a policy with an invalidation (INVAL) action and a content group that uses an invalidation selector or parameters to selectively expire cached objects in the group.
Full inval requests	Requests that match an invalidation policy where the invalGroups parameter is configured and expires one or more content groups.
Inval requests	Requests that match an invalidation policy and result in expiration of specific cached responses or entire content groups.
Parameterized requests	Number of cache requests that were processed using a policy with a parameterized content group.
Parameterized non-304 hits	Number of cache requests that were processed using a policy with a parameterized content group, where full cached response was found, and the response was not a 304 (object not updated) response.
Parameterized 304 hits	Number of cache requests that were processed using a policy with a parameterized content group, where the cached object was found, and the object was a 304 (object not updated) response.
Total parameterized hits	Number of cache requests that were processed using a policy with a parameterized content group, where the cached object was found.

Counter	Specifies
Parameterized 304 hit ratio (%)	Percentage of 304 (object not updated) responses that were found using a parameterized policy, relative to all cache hits.
Poll every time requests	If Poll Every Time is enabled, the NetScaler appliance always consults the origin server before serving a stored object. For more information, see "Polling the Origin Server Every Time a Request Is Received."
Poll every time hits	Number of times a cache hit was found using the Poll Every Time method. For more information, see "Polling the Origin Server Every Time a Request Is Received."
Poll every time hit ratio (%)	Percentage of cache hits using the Poll Every Time method, relative to all searches for cached objects using Poll Every Time. For more information, see "Polling the Origin Server Every Time a Request Is Received."
Maximum memory (KB)	Maximum amount of memory in the NetScaler appliance that is allocated to the cache. For more information, see "Configuring Global Attributes for Caching."
Maximum memory active value (KB)	Maximum amount of memory (active value) that will be set after the memory is actually allocated to the cache. For more information, see "How to Configure the Integrated Caching Feature of a NetScaler Appliance for various Scenarios."
Utilized memory (KB)	Amount of memory that is actually being used.
Memory allocation failures	Number of failed attempts to utilize memory for the purpose of storing a response in the cache.
Largest response so far	Largest response in bytes found in either the cache or the origin server and sent to the client.
Cached objects	Number of objects in the cache, including responses that have not yet been fully downloaded and responses that have been expired but not yet flushed.
Marker objects	Marker objects are created when a response exceeds the maximum or minimum response size for the content group, or has not yet received the minimum number of hits for the content group.

Counter	Specifies
Hits being served	Number of hits that have been served from the cache.
Misses being handled	Responses that were fetched from the origin server, stored in the cache, and then served. Should approximate the number for storable misses. Does not include non-storable misses.

To view summary cache statistics by using the command line interface

At the command prompt, type:

stat cache

To view specific cache statistics by using the command line interface

At the command prompt, type:

stat cache -detail [-fullValues] [-ntimes <positiveInteger>] [-logFile <inputFilename>]

To view summary cache statistics by using the configuration utility

1. Click the Dashboard tab at the top of the page.
2. Scroll down to the Integrated Caching section of the window.
3. To see detailed statistics, click the More... link at the bottom of the table.

To view specific cache statistics by using the configuration utility

1. Click the Reporting tab at the top of the page.
2. Under Built-In Reports, expand Integrated Cache, and then click the report with the statistics you want to view.
3. To save the report as a template, click Save As and name the report. The saved report appears under Custom Reports.

Improving Cache Performance

May 19, 2015

You can improve the performance of integrated cache, including handling simultaneous requests for the same cached data, avoiding delays that are associated with refreshing cached responses from the origin server, and ensuring that a response is requested often enough to be worth caching.

This section includes the following details:

- Reducing Flash Crowds
- Caching a Response after a Client Halts a Download
- Requiring a Minimum Number of Server Hits before Caching
- Example of Performance Optimization

Reducing Flash Crowds

Updated: 2015-05-20

Flash crowds occur when many users simultaneously request the same data. All of the requests in a flash crowd can become cache misses if you configured the cache to serve hits only after the entire object is downloaded.

The following techniques can reduce or eliminate flash crowds:

- **PREFETCH**: Refreshes a positive response before it expires to ensure that it never becomes stale or inactive.
  For more information, see "Refreshing a Response Prior to Expiration."
- **Cache buffering**: Starts serving a response to multiple clients as soon as it receives the response header from the origin server, rather than waiting for the entire response to be downloaded.
  The only limit on the number of clients that can download a response simultaneously is the available system resources.

  The Citrix NetScaler appliance downloads and serves responses even if the client that initiated the download halts before the download is complete. If the size of the response exceeds the cache size or if the response is chunked, the cache stops storing the response, but service to the clients is not disrupted.
- **Flash Cache**: Flash Cache queues requests to the cache, and allows only one request to reach the server at a time.
  For more information, see "Queuing Requests to the Cache."

Refreshing a Response Before Expiration

To ensure that a cached response is fresh whenever it is needed, the PREFETCH option refreshes a response before its calculated expiration time. The prefetch interval is calculated after receiving the first client request. From that point onward, the NetScaler appliance refreshes the cached response at a time interval that you configure in the PREFETCH parameter.

This setting is useful for data that is updated frequently between requests. It does not apply to negative responses (for example, 404 messages).

To configure prefetch for a content group by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -prefetch YES [-prefetchPeriod <seconds> | -prefetchPeriodMilliSec <milliseconds>] [-

prefetchMaxPending <positiveInteger>]
To configure prefetch for a content group by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Others tab, in the Flash Crowd and Prefetch group, select Prefetch option, and specify the values in Interval and Maximum number of pending prefetches text boxes.

Queuing Requests to the Cache

The Flash Cache option queues requests that arrive simultaneously (a flash crowd), retrieves the response, and distributes it to all the clients whose requests are in the queue. If, during this process, the response becomes non-cacheable, the NetScaler appliance stops serving the response from the cache and instead serves the origin server's response to the queued clients. If the response is not available, the clients receive an error message.

Flash Cache is disabled by default. You cannot enable Poll Every Time (PET) and Flash Cache on the same content group.

One disadvantage of Flash Cache is if the server replies with an error (for example, a 404 that is quickly remedied), the error is fanned out to the waiting clients.

Note: If Flash Cache is enabled, in some situations the NetScaler appliance is unable to correctly match the Accept-Encoding header in the client request with the Content-Encoding header in the response. The NetScaler appliance can assume that these headers match and mistakenly serve a hit. As a work-around, you can configure Integrated Caching policies to disallow serving hits to clients that do not have an appropriate Accept-Encoding header.
To enable Flash Cache by using the command line interface

At the command prompt, type:

set cache contentgroup <contentGroupName> -flashcache yes
To enable Flash Cache by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Others tab, in the Flash Crowd and Prefetch group, select Prefetch option.

Caching a Response after a Client Halts a Download

Updated: 2014-08-08

You can set the Quick Abort parameter to continue caching a response, even if the client halts a request before the response is in the cache.

If the downloaded response size is less than or equal to the Quick Abort size, the NetScaler appliance stops downloading the response. If you set the Quick Abort parameter to 0, all downloads are halted.

To configure quick abort size by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -quickAbortSize <integerInKBytes>

To configure quick abort size by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Memory tab, set the relevant value in Quick Abort: Continue caching if more than text box.

Requiring a Minimum Number of Server Hits before Caching

Updated: 2015-05-19

You can configure the minimum number of times that a response must be found on the origin server before it can be cached. You should consider increasing the minimum hits if the cache memory fills up quickly and has a lower-than-expected hit ratio.

The default value for the minimum number of hits is 0. This value caches the response after the first request.

To configure the minimum number of hits that are required before caching by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -minhits <positiveInteger>

To configure the minimum number of hits that are required before caching by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Memory tab, set the relevant value in Do not cache, if hits are less than text box.

Example of Performance Optimization

Updated: 2013-10-28

In this example, a client accesses a stock quote. Stock quotes are highly dynamic. You configure the integrated cache to serve the same stock quote to concurrent clients without sending multiple requests to the origin server. The stock quote expires after it is downloaded to all of the clients, and the next request for a quote for the same stock is fetched from the origin server. This ensures that the quote is always up to date.

The following task overview describes the steps to configure the cache for the stock quote application.

Task overview: Configuring caching for a stock quote application

1. Create a content group for stock quotes.
   For more information, see "About Content Groups."

   Configure the following for this content group:

   - On the Expiry Method tab, select the Expire after complete response received check box.
   - On the Others tab, select the Flash Cache check box, and click Create.
2. Add a cache policy to cache the stock quotes.
   For more information, see "Configuring a Policy in the Integrated Cache."

   Configure the following for the policy:

   - In the Action and Store in Group lists, select CACHE and select the group that you defined in the previous step.
   - Click Add, and in the Add Expression dialog box configure an expression that identifies stock quote requests, for example:
     `http.req.url.contains("cgi-bin/stock-quote.pl")`

3. Activate the policy.
   For more information, see "Globally Binding an Integrated Caching Policy." In this example, you bind this policy to request-time override processing and set the priority to a low value.

Configuring Cookies, Headers, and Polling

May 20, 2015

This section describes the procedures to configure how the cache manages cookies, HTTP headers, and origin server polling, including modifying default behavior that causes the cache to diverge from documented standards, overriding HTTP headers that might cause cacheable content to not be stored in the cache, and configuring the cache to always poll the origin for updated content under specialized circumstances.

For details, see the following sections:

- Divergence of Cache Behavior from the Standards
- Removing Cookies from a Response
- Inserting HTTP Headers at Response Time
- Ignoring Cache-Control and Pragma Headers in Requests
- Polling the Origin Server Every Time a Request Is Received
- PET and Client-Specific Content
- PET and Authentication, Authorization, and Auditing

Divergence of Cache Behavior from the Standards

Updated: 2013-10-28

By default, the integrated cache conforms to the following standards:

- RFC 2616, "Hypertext Transfer Protocol HTTP/1.1"
- The caching behaviors described in RFC 2617, "HTTP Authentication: Basic and Digest Access Authentication"
- The caching behavior described in RFC 2965, "HTTP State Management Mechanism"

The built-in policies and the Default content group attributes ensure conformance with most of these standards.

The default integrated cache behavior diverges from the specifications as follows:

- There is limited support for the Vary header.
  By default, any response containing a Vary header is considered to be non-cacheable unless it is compressed. A compressed response contains `Content-Encoding: gzip`, `Content-Encoding: deflate`, or `Content-Encoding: pack200-gzip` and is cacheable even if it contains the `Vary: Accept-Encoding` header.
- The integrated cache ignores the values of the headers `Cache-Control: no-cache` and `Cache-Control: private`.
  For example, a response that contains `Cache-Control: no-cache="Set-Cookie"` is treated as if the response contained `Cache-Control: no-cache`. By default, the response is not cached.
- An image (Content-Type = image/*) is always considered cacheable even if an image response contains Set-Cookie or Set-Cookie2 headers, or if an image request contains a Cookie header.
  The integrated cache removes Set-Cookie and Set-Cookie2 headers from a response before caching it. This diverges from RFC 2965. You can configure RFC-compliant behavior as follows:

  ```
  add cache policy rfc_compliant_images_policy -rule "http.res.header.set-cookie2.exists || http.res.header.set-cookie.exists" -action NOCACHE
  bind cache global rfc_compliant_images_policy -priority 100 -type REQ_OVERRIDE
  ```
- The following Cache-Control headers in a request force an RFC-compliant cache to reload a cached response from the origin server:

  ```
  Cache-control: max-age=0
  Cache-control: no-cache
  ```
  To guard against Denial of Service attacks, this behavior is not the default. For more information, see "Inserting a Cache-Control Header."
- By default, the caching module considers a response to be cacheable unless a response header states otherwise.
  To make this behavior RFC 2616 compliant, set -weakPosRelExpiry and -weakNegResExpiry to 0 for all content groups.

Removing Cookies from a Response

Updated: 2014-08-12

Cookies are often personalized for a user, and typically should not be cached. The Remove Response Cookies parameter removes Set-Cookie and Set-Cookie2 headers before caching a response. By default, the Remove Response Cookies option for a content group prevents caching of responses with Set-Cookie or Set-Cookie2 headers.

Note that when images are cached, the built-in behavior is to remove the Set-Cookie and Set-Cookie2 headers before caching, no matter how the content group is configured.

Note: Citrix recommends that you accept the default Remove Response Cookies for every content group that stores embedded responses, for example, images.

To configure Remove Response Cookies for a content group by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -removeCookies YES

To configure Remove Response Cookies for a content group by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Others tab, in the Settings group, select Remove response cookies option.

Inserting HTTP Headers at Response Time

Updated: 2014-08-12

The integrated cache can insert HTTP headers in responses that result from cache hits. The Citrix® NetScaler® appliance does not alter headers in responses that result from cache misses.

The following table describes headers that you can insert in a response.

Table 1. Different HTTP Headers You Can Insert in a Response That Is Served from the Cache

Header	Specifies
Age	Provides the age of the response in seconds, calculated from the time the response was generated at the origin server. By default, the cache inserts an Age header for every response that is served from the cache.
Via	Lists protocols and recipients between the start and end points for a request or a response. The NetScaler appliance inserts a Via header in every response that it serves from the cache. The default value of the inserted header is “NS-CACHE-9.2:last octet of the NetScaler IP address.” For more information, see "Configuring Global Attributes for Caching."
ETag	The cache supports response validation using Last-Modified and ETag headers to determine if a response is stale. The cache inserts an ETag in a response only if it caches the response and the origin server has not inserted its own ETag header. The ETag value is an arbitrary unique number. The ETag value for a response changes if it is refreshed from the origin server, but it stays the same if the server sends a 304 (object not updated) response. Origin servers typically do not generate validators for dynamic content because dynamic content is considered non-cacheable. You can override this behavior. With ETag header insertion, the cache is permitted to not serve full responses. Instead, the user agent is required to cache the dynamic response sent by the integrated cache the first time. To force a user agent to cache a response, you configure the integrated cache to insert an ETag header and replace the origin-provided Cache-Control header.
Cache-Control	The NetScaler appliance typically does not modify cacheability headers in responses that is serves from the origin server. If the origin server sends a response that is labeled as non-cacheable, the client treats the response as non-cacheable even if the NetScaler appliance caches the response. To cache dynamic responses in a user agent, you can replace Cache-Control headers from the origin server. This applies only to user agents and other intervening caches. They do not affect the integrated cache. For more information, see "Inserting a Cache-Control Header."

Inserting an Age, Via, or ETag Header

The following procedures describe how to insert Age, Via, and ETag headers.

To insert an Age, Via, or Etag header by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -insertVia YES -insertAge YES -insertETag YES
To insert an Age, Via, or Etag header by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Others tab, in the HTTP Header Insertions group, select the Via, Age, or ETag options, as appropriate.
   The values for the other header types are calculated automatically. Note that you configure the Via value in the main settings for the cache.

Inserting a Cache-Control Header

When the integrated cache replaces a Cache-Control header that the origin server inserted, it also replaces the Expires header. The new Expires header contains an expiration time in the past. This ensures that HTTP/1.0 clients and caches (that do not understand the Cache-Control header) do not cache the content.

To insert a Cache-Control header by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -cacheControl <value>
To insert a Cache-Control header by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and
   1. Click Expiry Method tab, clear the heuristic and default expiry settings and set the relevant value in Expire content after text box.
   2. Click Others tab and type the header you want to insert in the Cache-Control text box. Alternatively, click Configure to set the Cache-Control directives in cached responses.

Ignoring Cache-Control and Pragma Headers in Requests

Updated: 2014-08-12

By default, the caching module processes Cache-Control and Pragma headers. The following tokens in Cache-Control headers are processed as described in RFC 2616.

- max-age
- max-stale
- only-if-cached
- no-cache

A Pragma: no-cache header in a request is treated in the same way as a Cache-Control: no-cache header.

If you configure the caching module to ignore Cache-Control and Pragma headers, a request that contains a Cache-Control: No-Cache header causes the NetScaler appliance to retrieve the response from the origin server, but the cached response is not updated. If the caching module processes Cache-Control and Pragma headers, the cached response is refreshed.

The following table summarizes the implications of various settings for these headers and the Ignore Browser's Reload Request setting.

Table 2. Outcome of Settings for Ignoring Reload Requests, Cache-Control, and Pragma Headers

Setting for Ignore Cache-Control and Pragma Headers	Setting for Ignore Browser's Reload Request	Outcome
Yes	Yes or No	Ignore the Cache-Control and Pragma headers from the client, including the Cache-Control: no-cache directive.
No	Yes	The Cache-Control: no-cache header produces a cache miss, but a response that is already in the cache is not refreshed.
No	No	A request that contains a Cache-Control: no-cache header causes a cache miss and the stored response is refreshed.

To ignore Cache-Control and Pragma headers in a request by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -ignoreReqCachingHdrs YES

To ignore browser reload requests by using the command line interface

At the command prompt, type:

set cache contentgroup <name> -ignoreReloadReq NO
Note that by default, the -ignoreReloadReq parameter is set to YES.

To ignore Cache-Control and Pragma headers in a request by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Others tab, in the Settings group, select Ignore Cache-control and Pragma Headers in Requests option.

Example of a Policy to Ignore Cache-Control Headers

In the following example, you configure a request-time override policy to cache responses that contain Content-type: image/* regardless of the Cache-Control header in the response.

To configure a request-time override policy to cache all responses with image/*

1. Flush the cache using the Invalidate All option.

   For more information, see "Flushing Responses in a Content Group."

2. Configure a new cache policy, and direct the policy to a particular content group. For more information, see "Configuring a Policy in the Integrated Cache."
3. Ensure the content group that the policy uses is configured to ignore Cache-Control headers, as described in "Ignoring Cache-Control and Pragma Headers in Requests."
4. Bind the policy to the request-time override policy bank.

   For more information, see "Globally Binding an Integrated Caching Policy."

Polling the Origin Server Every Time a Request Is Received

Updated: 2014-08-12

You can configure the NetScaler appliance to always consult the origin server before serving a stored response. This is known as Poll Every Time (PET). When the NetScaler appliance consults the origin server and the PET response has not expired, a full response from the origin server does not overwrite cached content. This property is useful when serving client-specific content.

After a PET response expires, the NetScaler appliance refreshes it when the first full response arrives from the origin server.

The Poll Every Time (PET) function works as follows:

- For a cached response that has validators in the form of an ETag or a Last-Modified header, if the response expires it is automatically marked PET and cached.
- You can configure PET for a content group.
  If you configure a content group as PET, every response in the content group is marked PET. The PET content group can store responses that do not have validators. Responses that are automatically marked PET are always expired. Responses that belong to a PET content group can expire after a delay, based on how you configure the content group.

Two types of requests are affected by polling:

- **Conditional Requests**: A client issues a conditional request to ensure that the response that it has is the most recent copy.
  A user-agent request for a cached PET response is always converted to a conditional request and sent to the origin server. A conditional request has validators in If-Modified-Since or If-None-Match headers. The If-Modified-Since header contains the time from the Last-Modified header. An If-None-Match header contains the response's ETag header value.

  If the client's copy of the response is fresh, the origin server replies with 304 Not Modified. If the copy is stale, a conditional response generates a 200 OK that contains the entire response.

- **Non-Conditional Requests**: A non-conditional request can only generate a 200 OK that contains the entire response.

The following table summarizes response types based on the origin server's response

Table 3. How Responses Are Affected by Poll Every Time

Origin Server Response	Action
Send the full response	The origin server sends the response as-is to the client. If the cached response has expired, it is refreshed.
304 Not Modified	The following header values in the 304 response are merged with the cached response and the cached response is served to the client: • Date • Expires • Age • Cache-Control header Max-Age and S-Maxage tokens
401 Unauthorized 400 Bad Request 405 Method Not Allowed 406 Not Acceptable	The origin's response is served as-is to the client. The cached response is not changed.

Origin Server Response	Action
407 Proxy Authentication Required	
Any other error response, for example, 404 Not Found	The origin's response is served as-is to the client. The cached response is removed.

Note: The Poll Every Time parameter treats the affected responses as non-storable.

To configure poll every time by using the command line interface

At the command prompt, type:

add cache contentgroup <contentGroupName> -pollEveryTime YES

To configure poll every time by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Content Groups, and select the content group.
2. On Others tab, in the Settings group, select Poll every time (validate cached content with origin for every request) option.

PET and Client-Specific Content

The PET function can ensure that content is customized for a client. For example, a Web site that serves content in multiple languages examines the Accept-Language request header to select the language for the content that it is serving. For a multi-language Web site where English is the predominant language, all English language content can be cached in a PET content group. This ensures that every request goes to the origin server to determine the language for the response. If the response is English, and the content has not changed, the origin server can serve a 304 Not Modified to the cache.

The following example shows commands to cache English responses in a PET content group, configure a named expression that identifies English responses in the cache, and configure a policy that uses this content group and named expression. Bold is used for emphasis:

```
add cache contentgroup EnglishLanguageGroup -pollEveryTime YES
add expression containsENExpression –rule "http.res.header(\"Content-Language\").contains(\"en\")"
add cache policy englishPolicy -rule containsENExpression -action CACHE -storeInGroup englishLanguageGroup
bind cache policy englishPolicy -priority 100 -precedeDefRules NO
```

PET and Authentication, Authorization, and Auditing

Outlook Web Access (OWA) is a good example of dynamically generated content that benefits from PET. All mail responses (*.EML objects) have an ETag validator that enables them to be stored as PET responses.

Every request for a mail response travels to the origin server, even if the response is cached. The origin server determines whether the requestor is authenticated and authorized. It also verifies that the response exists in the origin server. If all results are positive, the origin server sends a 304 Not Modified response.

Configuring the Integrated Cache as a Forward Proxy

Aug 08, 2014

The integrated cache can service as a forward proxy device that passes requests to other NetScaler appliances or to other types of cache servers. You configure the integrated cache as a forward proxy by identifying the IP addresses of the cache server or servers. After configuring the forward proxy, the NetScaler appliance sends requests that contain the configured IP address on to the cache server instead of involving the integrated cache.

To configure the NetScaler as a forward cache proxy by using the command line interface

At the command prompt, type:

add cache forwardProxy <IPAddress> <port>

To configure the NetScaler as a forward cache proxy by using the configuration utility

1. Navigate to Optimization > Integrated Caching > Forward Proxy, and add a forward proxy by specifying the IP address and port number.

Default Settings for the Integrated Cache

Aug 26, 2013

The Citrix NetScaler integrated cache feature provides built-in policies with default settings as well as initial settings for the Default content group. The information in this section defines the parameters for the built-in policies and Default content group.

Default Caching Policies

Updated: 2013-08-26

The integrated cache has built-in policies. The NetScaler appliance evaluates the policies in a particular order, as discussed in the following sections.

You can override these built-in policies with a user-defined policy that is bound to a request-time override or response-time override policy bank.

Note that if you configured policies prior to release 9.0 and specified the -precedeDefRules parameter when binding the policies, they are automatically assigned to override-time bind points during migration.

Viewing the Default Policies

The built-in policy names start with an underscore (_). You can view the built-in policies from the command line and the administrative console using the show cache policy command.

Default Request Policies

You can override the following built-in request time policies by configuring new policies and binding them to the request-time override processing point. In the following policies, note that the MAY_NOCACHE action stipulates that the transaction is cached only when there is a user-configured or built-in CACHE directive at response time.

The following policies are bound to the _reqBuiltinDefaults policy label. They are listed in priority order.

1. Do not cache a response for a request that uses any method other than GET.
   The policy name is **_nonGetReq**. The following is the policy rule:

   ```
   !HTTP.REQ.METHOD.eq(GET)
   ```

2. Set a NOCACHE action for a request with header value that contains If-Match or If-Unmodified-Since.
   The policy name is **_advancedConditionalReq**. The following is the policy rule:

   ```
   HTTP.REQ.HEADER("If-Match").EXISTS || HTTP.REQ.HEADER("If-Unmodified-Since").EXISTS
   ```

3. Set a MAY_NOCACHE action for a request with the following header values: Cookie, Authorization, Proxy-authorization or a request which contains the NTLM or Negotiate header.
   The policy name is **_personalizedReq**. The following is the policy rule:

   ```
   HTTP.REQ.HEADER("Cookie").EXISTS || HTTP.REQ.HEADER("Authorization").EXISTS ||
   HTTP.REQ.HEADER("Proxy-Authorization").EXISTS || HTTP.REQ.IS_NTLM_OR_NEGOTIATE
   ```

Default Response Policies

You can override the following default response-time policies by configuring new policies and binding them to the response-time override processing point.

The following policies are bound to the **_resBuiltinDefaults** policy label and are evaluated in the order in which they are listed:

1. Do not cache HTTP responses unless they are of type 200, 304, 307, 203 or if the types are between 400 and 499 or between 300 and 302.
   The policy name is **_uncacheableStatusRes**. The following is the policy rule:

   !((HTTP.RES.STATUS.EQ(200)) || (HTTP.RES.STATUS.EQ(304)) || (HTTP.RES.STATUS.BETWEEN(400,499)) || (HTTP.RES.STATUS.BETWEEN(300, 302)) || (HTTP.RES.STATUS.EQ(307))|| (HTTP.RES.STATUS.EQ(203)))

2. Do not cache an HTTP response if it has a Vary header with a value of anything other than Accept-Encoding.
   The compression module inserts the Vary: Accept_Encoding header. The name of this expression is **_uncacheableVaryRes**. The following is the policy rule:

   ((HTTP.RES.HEADER("Vary").EXISTS) && ((HTTP.RES.HEADER("Vary").INSTANCE(1).LENGTH > 0) || (!HTTP.RES.HEADER("Vary").STRIP_END_WS.SET_TEXT_MODE(IGNORECASE).eq("Accept-Encoding"))))

3. Do not cache a response if its Cache-Control header value is No-Cache, No-Store, or Private, or if the Cache-Control header is not valid.
   The policy name is **_uncacheableCacheControlRes**. The following is the policy rule:

   ((HTTP.RES.CACHE_CONTROL.IS_PRIVATE) || (HTTP.RES.CACHE_CONTROL.IS_NO_CACHE) || (HTTP.RES.CACHE_CONTROL.IS_NO_STORE) || (HTTP.RES.CACHE_CONTROL.IS_INVALID))

4. Cache responses if the Cache-Control header has one of the following values: Public, Must-Revalidate, Proxy-Revalidate, Max-Age, S-Maxage.
   The policy name is **_cacheableCacheControlRes**. The following is the policy rule:

   ((HTTP.RES.CACHE_CONTROL.IS_PUBLIC) || (HTTP.RES.CACHE_CONTROL.IS_MAX_AGE) || (HTTP.RES.CACHE_CONTROL.IS_MUST_REVALIDATE) || (HTTP.RES.CACHE_CONTROL.IS_PROXY_REVALIDATE) || (HTTP.RES.CACHE_CONTROL.IS_S_MAXAGE))

5. Do not cache responses that contain a Pragma header.
   The name of the policy is **_uncacheablePragmaRes**. The following is the policy rule:

   HTTP.RES.HEADER("Pragma").EXISTS

6. Cache responses that contain an Expires header.
   The name of the policy is **_cacheableExpiryRes**. The following is the policy rule:

   HTTP.RES.HEADER("Expires").EXISTS

7. If the response contains a Content-Type header with a value of Image, remove any cookies in the header and cache it.
   The name of the policy is **_imageRes**. The following is the policy rule:

   HTTP.RES.HEADER("Content-Type").SET_TEXT_MODE(IGNORECASE).STARTSWITH("image/")

   You could configure the following content group to work with this policy:

```
add cache contentgroup nocookie_group -removeCookies YES
```

8. Do not cache a response that contains a Set-Cookie header.
   The name of the policy is **_ personalizedRes**. The following is the policy rule:

   ```
   HTTP.RES.HEADER("Set-Cookie").EXISTS || HTTP.RES.HEADER("Set-Cookie2").EXISTS
   ```

Restrictions on Default Policies

You cannot override the following built-in request time policies with user-defined policies.

These policies are listed in priority order.

1. Do not cache any responses if the corresponding HTTP request lacks a GET or POST method.
2. Do not cache any responses for a request if the HTTP request URL length plus host name exceeds 1744 byes.
3. Do not cache a response for a request that contains an If-Match header.
4. Do not cache a request that contains an If-Unmodified-Since header.
   Note that this is different from the If-Modified-Since header.
5. Do not cache a response if the server does not set an expiry header.

You cannot override the following built-in response time policies. These policies are evaluated in the order in which they are listed:

1. Do not cache responses that have an HTTP response status code of 201, 202, 204, 205, or 206.
2. Do not cache responses that have an HTTP response status code of 4xx, with the exceptions of status codes 403, 404, and 410.
3. Do not cache responses if the response type is FIN terminated, or the response does not have one of the following attributes: Content-Length, or Transfer-Encoding: Chunked.
4. Do not cache the response if the caching module cannot parse its Cache-Control header.

Initial Settings for the Default Content Group

When you first enable integrated caching, the NetScaler appliance provides one predefined content group named the Default content group. The following table shows the settings for this group.

Table 1. Predefined Settings for the Default Content Group

Parameter	Description	Default Value
Hit parameters	The hit parameters contain the parameter names that are significant for generating a response. In parameterized hit selection, NetScaler appliance matches the URL stem byte-for-byte, matches normalized values of the hit parameters, and matches the target service information.	none
Invalidation Parameters	These parameters mark a cached object as obsolete during parameterized selection. Specific objects, or all objects in a content group, are selected if the values of the	none

Parameter	Description	Default Value
	invalidation parameters in the object and in the request are same after normalization. The invalidation parameters are a subset of the hit parameters.	
Poll Every Time	Poll every time for the objects in this content group.	NO
Ignore reload request	Specifies whether a request can force the system to reload a cached object from the origin. To guard against Denial of Service attacks, you must set this flag to YES. To get RFC-compliant behavior you should set it to NO.	YES
Remove Response Cookies	If this option is disabled for a content group, and if the response contains cookies, the cookies are stored and served with every cache hit. By default, the remove cookies option is enabled for a content group, to prevent the integrated cache from storing any responses with Set-Cookie or Set-Cookie2 headers unless the response is an image.	YES
Prefetch	The Prefetch option refreshes an object when it is about to expire. This ensures that the object remains stale or inactive (and therefore it cannot be served) for a shorter duration of time.	YES
Prefetch period	This duration in seconds during which prefetch should be attempted, immediately before the object's calculated expiry time.	heuristic
Maximum outstanding prefetches	The number of items that can be subjected to a prefetch at a time.	4294967295
Flashcache	Determines whether to enable queuing of client requests and simultaneous distribution of responses to all clients in the queue.	NO
Expire at last byte	Determines whether to expire a cached response immediately after serving it.	NO
Insert Via header	Defines a string to be inserted in a Via header. By default, a Via header is inserted in all responses served from a content group. The Via header is not inserted for responses that are served by the origin server.	YES
Insert Age header	The Age header contains information about the age of the object in seconds as calculated by the integrated cache.	YES
Insert ETag header	With ETag header insertion, the integrated cache does not serve full responses on repeat requests. This is done by forcing the user agent to cache the dynamic response	YES

Parameter	Description	Default Value
	sent by the cache the first time.	
Cache-control header	You can enable caching of dynamic objects in the user agent by replacing the Cache-Control headers that are inserted by the origin server. You must configure the new Cache-Control header to be inserted in the content group.	NONE
Quick abort size	If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.	4194303 KBytes (maximum)
Minimum Response Size	You can control memory use by setting a minimum response size. Cached objects must be larger than the minimum response size.	0 KBytes
Maximum Response Size	You can control memory use by setting a maximum response size. Cached objects must be smaller than the maximum response size.	80 KBytes
Memory usage limit	Sets the maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance. The minimum value is 0 and the maximum value is unlimited.	UNLIMITED
Ignore caching headers in request	Disregards Cache-Control and Pragma headers in HTTP requests.	YES
MinHits configured	Number of hits that are required to qualify a response for storage in this content group.	0
Always evaluate policies		NO
Pinned	By default, when the cache is full the NetScaler appliance replaces the least recently used response first. The NetScaler appliance does not apply this behavior to content groups that are marked as pinned.	NO
Lazy DNS resolution	If set to YES, DNS resolution is performed for responses only if the destination IP address in the request does not match the destination IP address of the cached response.	YES

Troubleshooting

Jul 22, 2013

If the integrated cache feature does not work as expected after you have configured it, you can use some common tools to access NetScaler resources and diagnose the problem.

Resources for Troubleshooting

Updated: 2013-07-22

For best results, use the following resources to troubleshoot an integrated cache issue on a NetScaler appliance:

- The relevant trace files
- The ns.conf file
- The RFC 2616 document
- A copy of the object, if possible

In addition to the above resources, the following tools expedite troubleshooting:

- The iehttpheaders or a similar utility
- The Wireshark application customized for the NetScaler trace files

Troubleshooting Integrated Caching Issues

Updated: 2013-08-02

The following are effective steps to troubleshoot the objects that are not cached:

1. **Verify that the Integrated Caching feature is enabled.**
   Run the following command to verify the feature is enabled:

   ```
   show ns feature
   ```

   Following is sample output of the above command:

   ```
   show ns feature
       Feature status:
                   Web Logging: OFF
             Surge Protection: OFF
               Load Balancing: ON
           Content Switching: ON
           Cache Redirection: OFF
                Sure Connect: OFF
          Compression Control: OFF
             Priority Queuing: OFF
              SSL Offloading: OFF
    Global Server Load Balancing: OFF
          Http DoS Protection: OFF
            Dynamic Routing: OFF
          Content Filtering: OFF
         Integrated Caching: ON
                  SSL VPN: OFF
   ```

OSPF Routing: OFF
RIP Routing: OFF
BGP Routing: OFF

Done

The entry highlighted in boldface (for reference) in the above output indicates that the integrated caching feature is enabled. If the feature is not enabled, run the following command to enable it:

enable ns feature IC

2. **Make sure that sufficient memory is available on the NetScaler appliance.**
   Depending on the size of the object to be cached, memory available to store the cacheable object might be insufficient. You can set the memory limit for the integrated cache either globally or for individual content groups.

   Run the following command to verify the memory allocated to integrated cache globally:

   show cache parameter

   Following is sample output of the above command:

   show cache parameter
   Integrated cache global configuration:
   Memory usage limit: 256 MBytes
   Via header: NS-CACHE-6.1: 101
   Verify cached object using: HOSTNAME_AND_IP
   Max POST body size to accumulate: 32768
   Current outstanding prefetches: 0
   Max outstanding prefetches: 4294967294
   Treat NOCACHE policies as BYPASS policies: YES

   Done

   The entry highlighted in boldface (for reference) in the preceding output indicates the amount of memory allocated to the integrated cache globally.

   Run the following command to verify the memory allocated to an individual content group:

   show cache contentGroup <Content_Group_Name>

   Following is sample output of the above command:

   show cache contentgroup content1
   Name: content1
   Heuristic expiry time parameter: 10 percent
   Weak relative expiry time - Positive responses: 3600 secs
   Weak relative expiry time - Negative responses: 600 secs
   Hit parameters: NONE
   Invalidation Parameters: NONE
   Invalidation restricted to host: NO
   Ignore parameter value case: NO
   Match Request Cookies: NO
   Poll Every Time: NO
   Ignore reload request: YES

```
Remove Response Cookies: YES
Prefetch: YES
Prefetch period: heuristic
Current outstanding prefetches: 0
Max outstanding prefetches: 4294967294
Flashcache: NO
Expire at last byte: NO
Insert Via header: YES
Insert Age header: YES
Insert ETag header: YES
Cache-control header: NONE
Quick abort size: 4194303 KBytes (MAXIMUM)
Minimum Response Size: 0 KBytes
Maximum Response Size: 80 KBytes
Memory usage: 0 Bytes
Memory usage limit: 64 MBytes
Ignore caching headers in request: YES
Non-304 hits: 0
304 hits: 0
Cached objects: 0
Number of times expired/flushed: 1
MinHits configured: 0
Always evaluate policies: NO
Pinned: NO
 Done
```

The entry highlighted in boldface (for reference) in the above output indicates the amount of memory allocated to the content group.

3. **Verify that cacheable object is small enough to be stored within the configured memory limits.**
   Complete the following procedure to make space for the object to be cached:
   - Run the following command to flush the cache for the content group to which the object belongs:
     `flush cache contentGroup <Content_Group_Name>`
   - Verify that the object is cached. If the object is cached successfully, increase the memory allocated for the content group. Otherwise, run the following command to flush the cache globally:
     `flush cache contentGroup ALL`
   - Verify that the object is cached. If the object is cached successfully, consider increasing the global memory limit. If the object is still not cached, something else is causing the failure to cache the object.

   The memory allocated to the integrated cache depends on the NetScaler appliance model. You can allocate approximately half the available memory to the integrated cache. Similarly, the maximum about to memory you can allocate for a content group cannot be more than the memory allocated for global cache.

   To increase the global memory limit for the integrated cache, run the following command:

   `set cache parameter -memLimit <Integer>`

   To increase the memory limit for a content group, run the following command:

set cache contentgroup <contentgroup name> -memLimit <Integer>

4. **Verify that the cache policy is bound to an appropriate bind point, an appropriate priority is set for the policy, and an appropriateprecedeDefRulesswitch is configured.**
   You must activate a caching policy by binding it globally. To verify that the policy is active, run the following command:

   show cache global

   Following is sample output of the above command:

   ```
   show cache global
   1)    Name: red_pol   State: ACTIVE   Priority: 1
         Rule: URL CONTAINS red
         Action: NOCACHE
         Precede default HTTP rules: YES
         Hits : 10
    Done
   ```

   In the output, verify the following settings:
   - **The policy is bound**: The output should contain all the active cache policies. If the cache policy for the object to be cached is not listed in the output, the policy is not yet bound. Run the following command to bind the policy globally:
     bind cache global <Policy_Name> -priority <Integer> [-precedeDefRules YES|NO]
   - **The policy is Active**: If the policy is bound, verify that the state of the policy is displayed as Active. The entry indicating that the policy in the preceding output is active is the first highlighted entry in the sample output of the show cache global command, above. The policy is active if it is bound globally and an appropriate priority is set. Otherwise, the status of the policy is showed as Passive.
   - **An appropriate priority is assigned to the policy**: The first highlighted entry in the sample output above displays the priority of the policy. If the priority is not set, you can use the bind command to set the priority of the policy. Note that the higher the priority number, the lower the priority. The priority assigned to the policy enables the NetScaler appliance to determine the order in which the policy should be evaluated.
     If evaluation of a particular policy fails, increase the priority of the policy so that it is evaluated before other policies. Caching policies, due to their high granularity, can be very complicated to configure. Therefore, two policies might be contradictory. As a result, only the higher-priority policy takes effect.
   - **The precedeDefRules switch** setting is correct: The second highlighted entry in the sample output of the show cache global command, above, indicates the precedeDefRules switch setting. This setting enables the NetScaler appliance to determine whether the policy should be evaluated before the default built-in policies, which implement the standard HTTP caching behavior, such as basing caching decisions on HTTP header fields (for example, the If-Modified-Since and no-cache fields). You can set this switch when binding the policy.
     For certain types of HTTP(S) transactions, you might have to make sure that the policy precedes default HTTP rules, to force objects to be cached. Especially if requests include header fields, such as If-Modified-Since, or responses contain the No-Cache header field, you might have to make sure that the cache policy overrides the default in order for objects from these transactions to be cached. Force the policy to override default HTTP rules by rebinding the cache policy with the -precedeDefRules YES switch.

5. **Verify the size of the object to be cached.**
   You can configure a content group with minimum, which by default is 0 KB, and maximum, which by default is 80 KB, response sizes for the objects to be cached. The object does not get cached if its size is not within the configured range. Additionally, verify that the cache expiry times are set to an appropriate value. For example, check for a very small

time limit, such as one second.

Run the following command from the command line interface of the appliance to display the size limits and expiry times for a specific content group:

```
show cache contentGroup <Content_Group_Name>
```

Following is an example of this command's output:

```
show cache contentGroup content1
    Name: content1
    Heuristic expiry time parameter: 10 percent
    Weak relative expiry time - Positive responses: 3600 secs
    Weak relative expiry time - Negative responses: 600 secs
    Hit parameters: NONE
    Invalidation Parameters: NONE
    Invalidation restricted to host: NO
    Ignore parameter value case: NO
    Match Request Cookies: NO
    Poll Every Time: NO
    Ignore reload request: YES
    Remove Response Cookies: YES
    Prefetch: YES
    Prefetch period: heuristic
    Current outstanding prefetches: 0
    Max outstanding prefetches: 4294967294
    Flashcache: NO
    Expire at last byte: NO
    Insert Via header: YES
    Insert Age header: YES
    Insert ETag header: YES
    Cache-control header: NONE
    Quick abort size: 4194303 KBytes (MAXIMUM)
    Minimum Response Size: 0 KBytes
    Maximum Response Size: 80 KBytes
    Memory usage: 0 Bytes
    Memory usage limit: UNLIMITED
    Ignore caching headers in request: YES
    Non-304 hits: 0
    304 hits: 0
    Cached objects: 0
    Number of times expired/flushed: 0
    MinHits configured: 0
    Always evaluate policies: NO
    Pinned: NO
 Done
```

In addition to the above steps for troubleshooting integrated caching issues, you can consider using the following troubleshooting techniques:

- Depending on the configuration of a policy, there are virtually an unlimited number of reasons for the policy not getting evaluated. If you have completed the preceding steps to troubleshoot the issue, consider completing the following procedure to troubleshoot the issue further:
  1. Flush the cache.
  2. Verify the value of the hit parameter for the policy by running the following command:
     show cache global
     1) Name: home_pol_1 State: ACTIVE Priority: 99
        Rule: URL CONTAINS home
        Action: NOCACHE
        Precede default HTTP rules: NO
        Hits : 29
  3. Send an HTTP request for the related object from a Web browser.
  4. Run the show cache global command again and verify that the value for the hit parameter has incremented.

  Depending on the policy receiving hits or not, you can determine whether the issue is due to the policy have not been configured correctly or to a more global cache setting.

Content Accelerator

Nov 27, 2015

> **Important**
>
> The content accelerator feature is no longer supported on the NetScaler appliance.

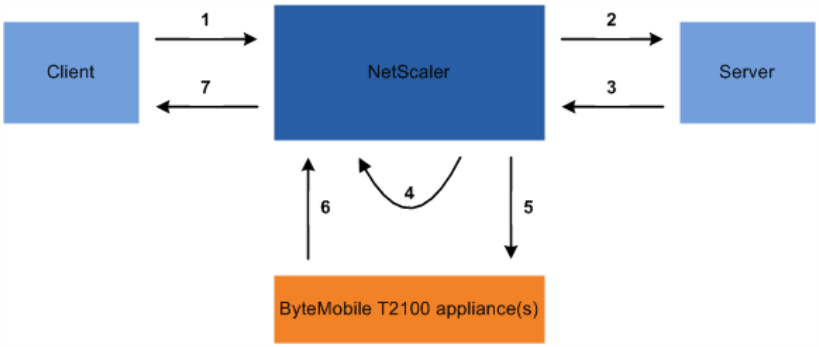
Content accelerator is a NetScaler feature that you can use in a Citrix ByteMobile T1100 deployment, to store data on a Citrix ByteMobile T2100 appliance. For more information about Citrix ByteMobile, see How ByteMobile Works.
Storing data on a T2100 appliance saves bandwidth and provides faster response times, because the NetScaler does not have to connect to the server for repeated requests of the same data.

Note: Content accelerator works with a Citrix ByteMobile platinum license. Contact customer support for more information and for obtaining the license.

How Content Accelerator Works

When a load balancing or content switching virtual server receives a client request, the NetScaler appliance evaluates a content accelerator policy that you have bound to the virtual server. The policy filters the requests to identify the ones to which to apply the content accelerator feature.

Note: For HTTP requests, the content accelerator feature can serve partial content in response to single byte-range requests.
The following figure illustrates the operations that the appliance performs when a client request arrives at a virtual server configured to use the content accelerator feature:



The process flow is as follows:

1. Client sends request.
2. NetScaler forwards the request to the server.
3. Server responds with the predefined size of the response (specified by the accumResSize parameter of the add ca action command).
4. NetScaler computes a hash of the response sent by the server.
5. NetScaler looks up the hash on the T2100 appliance.
6. A successful lookup indicates that the data is available and the T2100 appliance sends the data to the NetScaler.

   Note:
   - When a lookup does not succeed, the NetScaler fetches all of the requested data from the server, and simultaneously serves the data to the client and updates the data on the T2100 appliance.
   - The T2100 appliance can be configured to specify the number of requests after which to cache the data.
7. NetScaler sends the response to the client.

Configuring Content Accelerator

Before configuring the content accelerator feature, you must enable it on the NetScaler appliance.

You can configure the content accelerator feature to use one or multiple T2100 appliances. You must add each T2100 appliance as a service and bind these services to a load balancing virtual server that is dedicated to distributing the load between the configured T2100 appliances.

You must also configure a content accelerator action to lookup the data on the T2100 appliance. The action must also specify the T2100 load balancing virtual server and the size of data (in KB) to be fetched from the server to calculate the hash.

The action must be bound to a content accelerator policy that defines the traffic on which to perform content acceleration. The content accelerator policy must be bound to a content switching or load balancing virtual server that receives client traffic. Alternatively, you can bind the policy globally to be applicable to all virtual servers.

Configuring content accelerator by using the command line interface

At the command prompt, do the following:

1. Enable the content accelerator feature.
   enable ns feature ca

2. Identify the T2100 appliances and add each as a service on the NetScaler appliance.
   add service <name> <IPAddress> <serviceType> <port>

   Example:
   > add service T2100-A 10.102.29.61 HTTP 30
   > add service T2100-B 10.102.29.62 HTTP 40
   > add service T2100-C 10.102.29.63 HTTP 50
   Note: The services must be of type HTTP only.
3. Create a load balancing virtual server for the T2100 appliances. Specify the token load balancing method and the rule shown in the following syntax.
   add lb vserver <name> <serviceType> <IPAddress> <port> -lbMethod TOKEN -rule "http.req.url.after_str(\"/lookup/\") alt http.req.url.path.SKIP(1).PREFIX(64)"

   Example:
   > add lb vserver T2100-lbvserver HTTP 10.102.29.64 99 -lbMethod TOKEN -rule "http.req.url.after_str(\"/lookup/\") alt http.req.url.path.SKIP(1).PREFIX(64)"
4. Bind the T2100 services to the load balancing virtual server that you created for them.
   bind lb vserver <name> <serviceName>

   Example:
   > bind lb vserver T2100-lbvserver T2100-A
   > bind lb vserver T2100-lbvserver T2100-B
   > bind lb vserver T2100-lbvserver T2100-C
5. Define a content accelerator action.
   add ca action <name> -accumResSize <KBytes> -lbvserver <string> -type lookup

   Example:
   > add ca action ca_action1 -type lookup -lbvserver T2100-lbvserver -accumResSize 60
6. Define a content accelerator policy.
   add ca policy <name> -rule <expression> -action <name>

   Example: To create a content accelerator policy that caches all video formats.
   > add ca policy ca_mp4_pol -rule ns_video -action ca_action1
   where ns_video is a built-in expression.

7. Bind the content accelerator policy to either a virtual server that receives traffic or globally to the NetScaler system.

bind lb vserver <name> -policyName <string>

bind cs vserver <name> -policyName <string>

bind ca global -policyName <string> -priority <num> -type <type>

Example: To apply the content accelerator policy to a virtual server named "traf_rec"
> bind lb vserver traf_rec -policyName ca_mp4_pol
Example: To apply the content accelerator policy for all traffic reaching the NetScaler.
> bind ca global -policyName ca_mp4_pol -priority 100 -type RES_DEFAULT

8. Save the configuration.
   save ns config

Configuring content accelerator by using the configuration utility

1. Navigate to System > Settings > Configure Advanced Features and select Content Accelerator.
2. Create a service for each of the T2100 appliances.
   1. Navigate to Traffic Management > Load Balancing > Services.
   2. Click Add and specify the relevant details. In the Server field, make sure you specify the IP address of the T2100 appliance. In the Protocol field select HTTP.
3. Create a virtual server and bind the T2100 services to it.
   1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
   2. Click Add and specify the relevant details.
   3. In the Method and Persistence tab, specify the Method as Token.
   4. In the Policies tab, specify the rule as http.req.url.after_str(\"/lookup/\") alt http.req.url.path.SKIP(1).PREFIX(64).
   5. In the Services tab, select the T2100 services that you want to bind to the virtual server.
4. Create a content accelerator action.
   1. Navigate to Optimization > Content Accelerator > Actions.
   2. Specify the relevant details.
5. Create a content accelerator policy.
   1. Navigate to Optimization > Content Accelerator > Policies.
   2. Click Add, specify the policy rule, and associate the content accelerator action.
6. Bind the content accelerator policy globally or to a virtual server.
   1. Navigate to Optimization > Content Accelerator.
   2. Under the Content Accelerator Policy Manager [REQUEST] or Content Accelerator Policy Manager [RESPONSE] sections, bind the content accelerator policy globally or to a virtual server.

SPDY (Speedy)

May 20, 2015

Note: Supported from NetScaler 10.1 onwards.

SPDY is an open networking experimental protocol developed by Google to reduce the time taken by a client to load a web page in a browser. An application layer protocol, SPDY changes the way in which HTTP requests and responses are handled. SPDY offers the following advantages compared to a regular HTTP transaction:

- Multiplexed requests and responses—In a single SPDY session, multiple requests from the client can be sent over a single TCP connection to the server. This reduces the number of TCP connections and also optimizes usage of each TCP connection.
- Request prioritization—When requesting services from the server, a client can assign a priority to each request.
- Header Compression—SPDY compresses the HTTP request and response headers, saving bandwidth and reducing latency.
- Server push—The server can send data to the client before the client requests it.
- Security—SPDY is secure by design, because SSL is required for SPDY connections.

NetScaler supports the SPDY/2 and SPDY/3 (from NetScaler 10.5 onwards) versions.

Note: SPDY support depends on the browser version being used.

If you use a NetScaler appliance as a SPDY gateway for your servers, the servers do not have to support SPDY. The NetScaler appliance accepts the incoming SPDY requests, converts them, and sends them to the servers as HTTP requests. It also converts the HTTP responses and sends them to the clients as SPDY responses. While the key value of SPDY is reduced bandwidth consumption and faster communication with clients, an additional benefit of the NetScaler solution is that you avoid the time consuming task of upgrading your web servers and applications to support SPDY.

To use a NetScaler appliance as a SPDY gateway, you must enable SPDY on the appliance.

This document includes the following details:

- SPDY Requirements
- How SPDY Works over SSL
- Configuring SPDY on the NetScaler Appliance
- Troubleshooting for SPDY

SPDY Requirements

Both ends of a SPDY connection must support the same version of SPDY. In addition, the clients must meet the following requirements:

- Support ZLIB compression and accept compressed data.
- Support the Next Protocol Negotiation (NPN) TLS extension, because NPN is used in the TLS handshake.

How SPDY Works over SSL

Updated: 2014-03-13

When NetScaler sees an empty NPN extension in the Client Hello message, it responds with a list of the protocols that it supports. If SPDY is enabled on the NetScaler appliance, the appliance advertises HTTP/1.1 and SPDY/2 protocols. The client selects one protocol from this list and negotiates the protocol with the server. Because sending the negotiated protocol in plain text would raise security issues, the client sends the Change Cipher Spec notification which defines the details of the encryption for the session, followed by the Next Protocol message, which contains the encrypted protocol

that the client has chosen. The client then sends the Finished message. The NetScaler appliance decrypts the Next Protocol message, and then sends a Finished message.

A session is then established, and application data can be exchanged.

Note: The NPN extension is not supported on a NetScaler FIPS appliance, and with TLS protocol versions 1.1 and 1.2.

Configuring SPDY on the NetScaler Appliance

Updated: 2014-09-15

By default, SPDY is disabled on the NetScaler appliance. After you enable SPDY, the appliance advertises SPDY/2 and/or SPDY/3 along with HTTP/1.1 during an SSL handshake. To enable SPDY on the NetScaler appliance, you must enable SPDY in the HTTP profile bound to the SSL virtual server.

To configure SPDY by using the command line interface

At the command prompt, do the following:

1. Enable SPDY on a HTTP profile.
   set ns httpProfile <profileName> -SPDY <options>

   Example
   > set ns httpProfile profile1 -SPDY ENABLED
2. Bind the HTTP profile to a SSL virtual server.
   set lb vserver <ssl-vserver-name> -httpProfileName <httpProfile-with-spdy>

   Example
   > set lb vserver SPDY_LB -httpProfileName profile1
   Note: To apply SPDY globally, enable SPDY on the global HTTP profile (nshttp_default_profile).

You can view the statistics by using the following command:

stat protocol http -detail

To configure SPDY by using the configuration utility

1. Navigate to System > Profiles, and in the HTTP Profiles tab, update the profile on which you want to enable SPDY.
2. Navigate to Traffic Management > Load Balancing > Virtual Servers, and associate the HTTP profile to the appropriate SSL virtual server.

Troubleshooting for SPDY

If SPDY sessions are not enabled even after performing the required steps, check the following conditions.

- If the client is using a Chrome browser, SPDY might not work in some scenarios because Chrome sometimes does not initiate TLS handshake.
- If there is a forward-proxy between the client and the NetScaler appliance, and the forward-proxy doesn't support SPDY, SPDY sessions might not be enabled.
- NetScaler does not support NPN over TLS 1.1/1.2. To use SPDY, the client should disable TLS1.1/1.2 in the browser.
- Similarly, if the client wants to use SPDY, SSL2/3 must be disabled on the browser.

Security

May 03, 2013

The following topics cover configuration and installation information for NetScaler security features. Most of these features are policy based.

Authentication, Authorization, Auditing (AAA)	Keeps unauthorized users out of the network, denies users access to tasks for which they are not authorized, and tracks the resources used during user sessions.
Application Firewall	Prevents security breaches, data loss, and possible unauthorized modifications to web sites that access sensitive business or customer information.
Content Filtering	Blocks inappropriate HTML requests, preventing the requests from reaching the Web servers.
HTTP Denial-of-Service Protection	Prevents hackers from attacking your Web site with large numbers of HTTP requests.
Priority Queuing	Detects high-priority connections and allows those connections to proceed ahead of other connections, guaranteeing unimpeded access to those users.
SureConnect	Services all incoming connections with either the requested content or a custom Web page that displays information about a delay in the request being serviced.
Surge Protection	Detects any rapid rise in connection attempts and adjusts the rate at which connections are allowed to proceed to the server, preventing server overload.

AAA Application Traffic

Sep 18, 2013

Many companies restrict web site access to valid users only, and control the level of access permitted to each user. The AAA feature allows a site administrator to manage access controls with the NetScaler appliance instead of managing these controls separately for each application. Doing authentication on the appliance also permits sharing this information across all web sites within the same domain that are protected by the appliance.

The AAA feature supports authentication, authorization, and auditing for all application traffic. To use AAA, you must configure authentication virtual servers to handle the authentication process and traffic management virtual servers to handle the traffic to web applications that require authentication. You also configure your DNS to assign FQDNs to each virtual server. After configuring the virtual servers, you configure a user account for each user that will authenticate via the NetScaler appliance, and optionally you create groups and assign user accounts to groups. After creating user accounts and groups, you configure policies that tell the appliance how to authenticate users, which resources to allow users to access, and how to log user sessions. To put the policies into effect, you bind each policy globally, to a specific virtual server, or to the appropriate user accounts or groups. After configuring your policies, you customize user sessions by configuring session settings and binding your session policies to the traffic management virtual server. Finally, if your intranet uses client certs, you set up the client certificate configuration.

Before configuring AAA, you should be familiar with and understand how to configure load balancing, content switching, and SSL on the NetScaler appliance.

How AAA Works

Oct 21, 2015

AAA provides security for a distributed Internet environment by allowing any client with the proper credentials to connect securely to protected application servers from anywhere on the Internet. This feature incorporates the three security features of authentication, authorization, and auditing. Authentication enables the NetScaler ADC to verify the client's credentials, either locally or with a third-party authentication server, and allow only approved users to access protected servers. Authorization enables the ADC to verify which content on a protected server it should allow each user to access. Auditing enables the ADC to keep a record of each user's activity on a protected server.

To understand how AAA works in a distributed environment, consider an organization with an intranet that its employees access in the office, at home, and when traveling. The content on the intranet is confidential and requires secure access. Any user who wants to access the intranet must have a valid user name and password. To meet these requirements, the ADC does the following:

- Redirects the user to the login page if the user accesses the intranet without having logged in.
- Collects the user's credentials, delivers them to the authentication server, and caches them in a directory that is accessible through LDAP.
- Verifies that the user is authorized to access specific intranet content before delivering the user's request to the application server.
- Maintains a session timeout after which users must authenticate again to regain access to the intranet. (You can configure the timeout.)
- Logs the user accesses, including invalid login attempts, in an audit log.

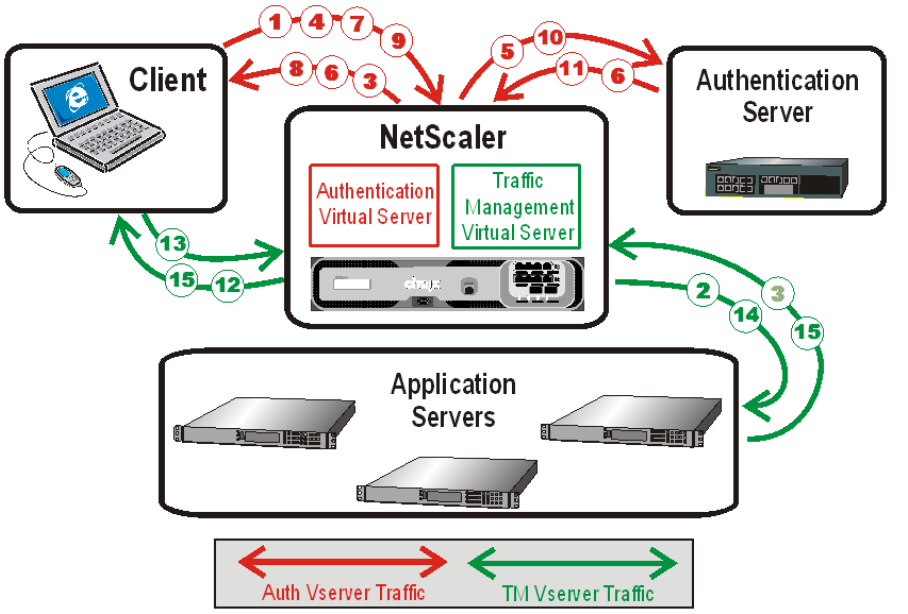
Authentication requires that several entities: the client, the NetScaler appliance, the external authentication server if one is used, and the application server, respond to each other when prompted by performing a complex series of tasks in the correct order. If you are using an external authentication server, this process can be broken down into the following fifteen steps.

1. The client sends a GET request for a URL on the application server.
2. The NetScaler appliance's traffic management virtual server redirects the request to the application server.
3. The application server determines that the client has not been authenticated, and therefore sends an HTTP 200 OK response via the TM vserver to the client. The response contains a hidden script that causes the client to issue a POST request for /cgi/tm.
4. The client sends a POST request for /cgi/tm.
5. The NetScaler appliance's authentication virtual server redirects the request to the authentication server.
6. The authentication server creates an authentication session, sets and caches a cookie that consists of the initial URL and the domain of the traffic management virtual server, and then sends an HTTP 302 response via the authentication virtual server, redirecting the client to /vpn/index.html.
7. The client sends a GET request for /vpn/index.html.
8. The authentication virtual server redirects the client to the authentication server login page.
9. The client sends a GET request for the login page, enters credentials, and then sends a POST request with the credentials back to the login page.
10. The authentication virtual server redirects the POST request to the authentication server.
11. If the credentials are correct, the authentication server tells the authentication virtual server to log the client in and redirect the client to the URL that was in the initial GET request.
12. The authentication virtual server logs the client in and sends an HTTP 302 response that redirects the client to the initially requested URL.

13. The client sends a GET request for their initial URL.
14. The traffic management virtual server redirects the GET request to the application server.
15. The application server responds via the traffic management virtual server with the initial URL.

If you use local authentication, the process is similar, but the authentication virtual server handles all authentication tasks instead of forwarding connections to an external authentication server. The following figure illustrates the authentication process.

Figure 1. Authentication Process Traffic Flow



When an authenticated client requests a resource, the ADC, before sending the request to the application server, checks the user and group policies associated with the client account, to verify that the client is authorized to access that resource. The ADC handles all authorization on protected application servers. You do not need to do any special configuration of your protected application servers.

AAA-TM handles password changes for users by using the protocol-specific method for the authentication server. For most protocols, neither the user nor the administrator needs to do anything different than they would without AAA-TM. Even when an LDAP authentication server is in use, and that server is part of a distributed network of LDAP servers with a single designated domain administration server, password changes are usually handled seamlessly. When an authenticated client of an LDAP server changes his or her password, the client sends a credential modify request to AAA-TM, which forwards it to the LDAP server. If the user's LDAP server is also the domain administration server, that server responds appropriately and AAA-TM then performs the requested password change. Otherwise, the LDAP server sends AAA-TM an LDAP_REFERRAL response to the domain administration server. AAA-TM follows the referral to the indicated domain administration server, authenticates to that server, and performs the password change on that server.

When configuring AAA-TM with an LDAP authentication server, the system administrator must keep the following conditions and limitations in mind:

- AAA-TM assumes that the domain administration server in the referral accepts the same bind credentials as the original server.
- AAA-TM only follows LDAP referrals for password change operations. In other cases AAA-TM refuses to follow the referral.
- AAA-TM only follows one level of LDAP referrals. If the second LDAP server also returns a referral, AAA-TM refuses to

follow the second referral.

The ADC supports auditing of all states and status information, so you can see the details of what each user did while logged on, in chronological order. To provide this information, the appliance logs each event, as it occurs, either to a designated audit log file on the appliance or to a syslog server. Auditing requires configuring the appliance and any syslog server that you use.

Enabling AAA

Oct 30, 2013

To use the AAA - Application Traffic feature, you must enable it. You can configure AAA entities—such as the authentication and traffic management virtual servers—before you enable the AAA feature, but the entities will not function until the feature is enabled.

At the command prompt, type the following commands to enable AAA and verify the configuration:

- enable ns feature AAA
- show ns feature

Example

```
> enable feature AAA
 Done

> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              OFF
 2)    Surge Protection          SP              ON
 .
 .
 .
 15)   AAA                      AAA              ON
 .
 .
 .
 23)   HTML Injection           HTMLInjection      ON
 24)   NetScaler Push           push             OFF
Done
```

1. Navigate to System > Settings.
2. In the details pane, under Modes and Features, click Change basic features.
3. In the Configure Basic Features dialog box, select the Authentication, Authorization and Auditing check box.
4. Click OK.

Setting up AAA Virtual Servers and DNS

Mar 28, 2012

You can configure AAA by using the built-in wizard, or manually. To use the wizard, in the main AAA pane of the configuration utility, you click AAA - Application Traffic wizard and follow the prompts.

To configure AAA manually, you first configure an authentication virtual server, which involves binding an SSL certificate-key pair. You then associate the authentication virtual server with a new or existing traffic management virtual server. (Either a load balancing virtual server or a content switching virtual server can serve as a traffic management virtual server.) To complete the initial configuration, you configure DNS to assign hostnames to both the authentication virtual server and the traffic management virtual server, and verify that your virtual servers are UP and configured correctly.

Caution: Both virtual servers must have hostnames in the same domain, or the AAA configuration will not work.

Configuring the Authentication Virtual Server

Aug 13, 2014

To configure AAA, first configure an authentication virtual server to handle authentication traffic. Next, bind an SSL certificate-key pair to the virtual server to enable it to handle SSL connections. For additional information about configuring SSL and creating a certificate-key pair, see the *Citrix NetScaler Traffic Management Guide* at "Traffic Management."

To configure an authentication virtual server and verify the configuration, at the command prompt type the following commands in the order shown:

- add authentication vserver <name> ssl <ipaddress>
- show authentication vserver <name>
- bind ssl certkey <certkeyName>
- show authentication vserver <name>
- set authentication vserver <name> -authenticationDomain <FQDN>
- show authentication vserver <name>

Example

```
> add authentication vserver Auth-Vserver-2 SSL 10.102.29.77 443
 Done
> show authentication vserver Auth-Vserver-2
      Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
      State: DOWN[Certkey not bound]
      Client Idle Timeout: 180 sec
      Down state flush: DISABLED
      Disable Primary Vserver On Down : DISABLED
      Authentication : ON
      Current AAA Users: 0
 Done
> bind ssl certkey Auth-Vserver-2 Auth-Cert-1
 Done
> show authentication vserver Auth-Vserver-2
      Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
      State: UP
      Client Idle Timeout: 180 sec
      Down state flush: DISABLED
      Disable Primary Vserver On Down : DISABLED
      Authentication : ON
      Current AAA Users: 0
      Authentication Domain: myCompany.employee.com
 Done
> set authentication vserver Auth-Vserver-2 -AuthenticationDomain myCompany.employee.com
 Done
> show authentication vserver Auth-Vserver-2
      Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
      State: DOWN[Certkey not bound]
```

```
    Client Idle Timeout: 180 sec
    Down state flush: DISABLED
    Disable Primary Vserver On Down : DISABLED
    Authentication : ON
    Current AAA Users: 0
    Authentication Domain: myCompany.employee.com
Done
```

1. Navigate to Security > AAA - Application Traffic > Virtual Servers.
2. In the details pane, do one of the following:
   - To create a new authentication virtual server, click Add.
   - To modify an existing authentication virtual server, select the virtual server, and then click Edit.

   The Configuration dialog opens with the Basic Settings area expanded.
3. Specify values for the parameters as follows (asterisk indicates a required parameter):
   - Name*—name (Cannot be changed for a previously created virtual server)
   - IP Address*—ipaddress
   - Domain*—authenticationDomain
   - Failed login timeout—failedLoginTimeout (Seconds allowed before login fails and user must start login process again.)
   - Max login attempts—maxLoginAttempts (Number of login attempts allowed before user is locked out)

   Note: The authentication virtual server uses only the SSL protocol and port 443, so those options are greyed out. Any options that are not mentioned are not relevant and should be ignored.
4. Click Continue to display the Certificates area.
5. In the Certificates area, configure any SSL certificates you want to use with this virtual server.
   - To configure a CA certificate, click the arrow on the right of CA Certificate to display the CA Cert Key dialog box, select the certificate you want to bind to this virtual server, and click Save.
   - To configure a server certificate, click the arrow on the right of Server Certificate, and follow the same process as for CA certificate.
6. Click Continue to display the Advanced Authentication Policies area.
7. If you want to bind an advanced authentication policy to the virtual server, click the arrow on the right side of the line to display the Authentication Policy dialog box, choose the policy that you want to bind to the server, set the priority, and then click OK.
8. Click Continue to display the Basic Authentication Policies area.
9. If you want to create a basic authentication policy and bind it to the virtual server, click the plus sign to display the Policies dialog box, and follow the prompts to configure the policy and bind it to this virtual server.
10. Click Continue to display the 401-Based Virtual Servers area.
11. In the 401-Based Virtual Servers area, configure any load balancing or content switching virtual servers that you want to bind to this virtual server.
    - To bind a load balancing virtual server, click the arrow to the right of LB virtual server to display the LB Virtual Servers dialog box, and follow the prompts.
    - To bind a content switching virtual server, click the arrow to the right of CS virtual server to display the CS Virtual Servers dialog box, and follow the same process as to bind an LB virtual server.
12. If you want to create or configure a group, in the Groups area click the arrow to display the Groups dialog box, and follow the prompts.
13. Review your settings, and when you are finished, click Done. The dialog box closes. If you created a new authentication virtual server, it now appears in the Configuration window list.

Configuring a Traffic Management Virtual Server

Aug 19, 2014

After you have created and configured your authentication virtual server, you next create or configure a traffic management virtual server and associate your authentication virtual sever with it. You can use either a load balancing or content switching virtual server for a traffic management virtual server. For more information about creating and configuring either type of virtual server, see the *Citrix NetScaler Traffic Management Guide* at Traffic Management.

Note: The FQDN of the traffic management virtual server must be in the same domain as the FQDN of the authentication virtual server for the domain session cookie to function correctly.

You configure a traffic management virtual server for AAA by enabling authentication and then assigning the FQDN of the authentication server to the traffic management virtual server. You can also configure the authentication domain on the traffic management virtual server at this time. If you do not configure this option, the NetScaler appliance assigns the traffic management virtual server an FQDN that consists of the FQDN of the authentication virtual server without the hostname portion. For example, if domain name of the authentication vserver is **tm.xyz.bar.com**, the appliance assigns **xyz.bar.com**. as the authentication domain.

At the command prompt, type one of the following sets of commands to configure a TM virtual server and verify the configuration:

- set lb vserver <name> –authentication ON -authenticationhost <FQDN> [-authenticationdomain <authdomain>]
- show lb vserver <name>
- set cs vserver <name> –authentication ON -authenticationhost <FQDN> [-authenticationdomain <authdomain>]
- show cs vserver <name>

Example

```
> set lb vserver vs-cont-sw -Authentication ON -AuthenticationHost mywiki.index.com
 Done
> show lb vserver vs-cont-sw
      vs-cont-sw (0.0.0.0:0) - TCP    Type: ADDRESS
      State: DOWN
      Last state change was at Wed Aug 19 10:03:15 2009 (+410 ms)
      Time since last state change: 5 days, 20:00:40.290
      Effective State: DOWN
      Client Idle Timeout: 9000 sec
      Down state flush: ENABLED
      Disable Primary Vserver On Down : DISABLED
      No. of Bound Services :  0 (Total)      0 (Active)
      Configured Method: LEASTCONNECTION
      Mode: IP
      Persistence: NONE
      Connection Failover: DISABLED
      Authentication: ON      Host: mywiki.index.com
 Done
```

1. In the navigation pane, do one of the following.
   - Navigate to Traffic Management > Load Balancing > Virtual Servers.
   - Navigate to Traffic Management > Content Switching > Virtual Servers
     The AAA configuration process for either type of virtual server is identical.
   - In the details pane, select the virtual server on which you want to enable authentication, and then click Open.
   - In the Domain text box, type the authentication domain.
   - On the Advanced tab, select the Authentication check box.
   - In the Authentication Host text box, type the fully qualified domain name of the authentication virtual server.
   - Click OK. A message appears in the status bar, stating that the vserver has been configured successfully.

Configuring DNS

Sep 24, 2013

For the domain session cookie used in the authentication process to function correctly, you must configure DNS to assign both the authentication and the traffic management virtual servers to FQDNs in the same domain. For information about how to the configure DNS address records, see the *Citrix NetScalerTraffic Management Guide* at ".Traffic Management"

Verifying Your Setup for AAA

Sep 18, 2013

After you configure authentication and traffic management virtual servers and before you create user accounts, you should verify that both virtual servers are configured correctly and are in the UP state.

At the command prompt, type the following command:
show authentication vserver <name>

Example

```
> show authentication vserver Auth-Vserver-2
      Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
      State: UP
      Client Idle Timeout: 180 sec
      Down state flush: DISABLED
      Disable Primary Vserver On Down : DISABLED
      Authentication : ON
      Current AAA Users: 0
      Authentication Domain: myCompany.employee.com
 Done
```

1. Navigate to Security > AAA - Application Traffic > Virtual Servers.
2. Review the information in the AAA Virtual Servers pane to verify that your configuration is correct and your authentication virtual server is accepting traffic. You can select a specific virtual server to view detailed information in the details pane.

Configuring Users and Groups

Aug 19, 2014

After configuring the AAA basic setup, you create users and groups. You first create a user account for each person who will authenticate via the NetScaler appliance. If you are using local authentication controlled by the NetScaler appliance itself, you create local user accounts and assign passwords to each of those accounts.

You also create user accounts on the NetScaler appliance if you are using an external authentication server. In this case, however, each user account must exactly match an account for that user on the external authentication server, and you do not assign passwords to the user accounts that you create on the NetScaler. The external authentication server manages the passwords for users that authenticate with the external authentication server.

If you are using an external authentication server, you can still create local user accounts on the NetScaler appliance if, for example, you want to allow temporary users (such as visitors) to log in but do not want to create entries for those users on the authentication server. You assign a password to each local user account, just as you would if you were using local authentication for all user accounts.

Each user account must be bound to policies for authentication and authorization. To simplify this task, you can create one or more groups and assign user accounts to them. You can then bind policies to groups instead of individual user accounts.

At the command prompt, type the following commands to create a local AAA user account and verify the configuration:

- add aaa user <username> [–password <password>]
- show aaa user

Example

```
> add aaa user user-2 -password emptybag
 Done
> show aaa user
1)     UserName: user-1
2)     UserName: user-2
 Done
```

At the command prompt, type the following command and, when prompted, type the new password:
set aaa user <username>
Example

```
> set aaa user user-2
Enter password:
Done
```

1. Navigate to Security > AAA - Application Traffic > Users
2. In the details pane, do one of the following:
   - To create a new user account, click Add.

- To modify an existing user account, select the user account, and then click Open.

3. In the Create AAA User dialog box, in the User Name text box, type a name for the user.
4. If creating a locally authenticated user account, clear the External Authentication check box and provide a local password that the user will use to log on.
5. Click Create or OK, and then click Close. A message appears in the status bar, stating that the user has been configured successfully.

At the command prompt, type the following commands. Type the first command one time, and type the second command once for each user:

- add aaa group <groupname>
- show aaa group

Example

```
> add aaa group group-2
Done
> show aaa group
1)     GroupName: group-1
2)     GroupName: group-2
Done
```

- bind aaa group <groupname> -username <username>

Example

```
> bind aaa group group-2 -username user-2
 Done
> show aaa group group-2
       GroupName: group-2

            UserName: user-2
 Done
```

At the command prompt, unbind users from the group by typing the following command once for each user account that is bound to the group:
unbind aaa group <groupname> -username <username>
Example

```
> unbind aaa group group-hr -username user-hr-1
 Done
```

First remove all users from the group. Then, at the command prompt, type the following command to remove an AAA group and verify the configuration:

- rm aaa group <groupname>
- show aaa group

Example

```
> rm aaa group group-hr
 Done
> show aaa group
1)     GroupName: group-1
2)     GroupName: group-finance
 Done
```

1. Navigate to Security > AAA - Application Traffic > Groups
2. In the details pane, do one of the following:
   - To create a new group, click Add.
   - To modify an existing group, select the group, and then click Open.
3. If you are creating a new group, in the Create AAA Group dialog box, in the Group Name text box, type a name for the group.
4. On the Users tab, configure the users assigned to the group.
   1. To add a user to the group, select the user, and then click Add.
   2. To remove a user from the group, select the user, and then click Remove.
   3. To create a new user account and add it to the group, click New, and then follow the instructions in "To configure AAA local users by using the configuration utility."
5. Click Create or OK. The group that you created appears in the AAA Groups page.

Configuring AAA Policies

Sep 18, 2013

After you set up your users and groups, you next configure authentication policies, authorization policies, and audit policies to define which users are allowed to access your intranet, which resources each user or group is allowed to access, and what level of detail AAA will preserve in the audit logs. An authentication policy defines the type of authentication to apply when a user attempts to log on. If external authentication is used, the policy also specifies the external authentication server. Authorization policies specify the network resources that users and groups can access after they log on. Auditing policies define the audit log type and location.

You must bind each policy to put it into effect. You bind authentication policies to authentication virtual servers, authorization policies to one or more user accounts or groups, and auditing policies both globally and to one or more user accounts or groups.

When you bind a policy, you assign a priority to it. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer. In the NetScaler operating system, policy priorities work in reverse order: the higher the number, the lower the priority. For example, if you have three policies with priorities of 10, 100, and 1000, the policy assigned a priority of 10 is performed first, then the policy assigned a priority of 100, and finally the policy assigned an order of 1000. The AAA feature implements only the first of each type of policy that a request matches, not any additional policies of that type that a request might also match, so policy priority is important for getting the results you intend.

You can leave yourself plenty of room to add other policies in any order, and still set them to evaluate in the order you want, by setting priorities with intervals of 50 or 100 between each policy when you bind the policies. You can then add additional policies at any time without having to reassign the priority of an existing policy.

For additional information about binding policies on the NetScaler, see the *Citrix NetScaler Traffic Management Guide* at "Traffic Management."

Authentication Policies

Aug 19, 2014

The NetScaler ADC can authenticate users with local user accounts or by using an external authentication server. The appliance supports the following authentication types:

LOCAL

Authenticates to the NetScaler by using a password, without reference to an external authentication server. User data is stored locally on the NetScaler appliance.

RADIUS

Authenticate to an external Radius server.

LDAP

Authenticates to an external LDAP authentication server.

TACACS

Authenticates to an external Terminal Access Controller Access-Control System (TACACS) authentication server.

After a user authenticates to a TACACS server, the NetScaler ADC connects to the same TACACS server for all subsequent authorizations. When a primary TACACS server is unavailable, this feature prevents delays while the ADC waits for the first TACACS server to time out before resending the authorization request to the second TACACS server.

Note: When authenticating through a TACACS server, AAA-TM logs only successfully executed TACACS commands, to prevent the logs from showing TACACS commands that were entered by users who were not authorized to execute them.

CERT

Authenticates to the NetScaler appliance by using a client certificate, without reference to an external authentication server.

NEGOTIATE

Authenticates to a Kerberos authentication server. If there is an error in Kerberos authentication, NetScaler uses NTLM authentication.

SAML

Authenticates to a server that supports the Security Assertion Markup Language (SAML).

SAMLIDP

Configures the NetScaler ADC to serve as a Security Assertion Markup Language (SAML) Identity Provider (IdP).

WEB

Authenticates to a web server, providing the credentials that the web server requires in an HTTP request and analyzing the web server response to determine that user authentication was successful.

An authentication policy is comprised of an expression and an action. Authentication policies use NetScaler expressions.

After creating an authentication action and an authentication policy, bind it to an authentication virtual server and assign a priority to it. When binding it, also designate it as either a primary or a secondary policy. Primary policies are evaluated before secondary policies. In configurations that use both types of policy, primary policies are normally more specific policies while secondary policies are normally more general policies intended to handle authentication for any user accounts that do not meet the more specific criteria.

If you do not use LOCAL authentication, you need to add an explicit authentication action. To do this, at the command prompt, type the following command:

add authentication tacacsAction <name> -serverip <IP> [-serverPort <port>] [-authTimeout <positive_integer>] [...]

Example

```
> add authentication tacacsaction Authn-Act-1 -serverip 10.218.24.65 -serverport 1812                -authtimeout 15 -tacacsSecret "minotaur" -authorization OFF -accounting
```

To configure an existing authentication action, at the command prompt, type the following command:

set authentication tacacsAction <name> -serverip <IP> [-serverPort <port>] [-authTimeout <positive_integer>] [...]

Example

```
> set authentication tacacsaction Authn-Act-1 -serverip 10.218.24.65 -serverport 1812                -authtimeout 15 -tacacsSecret "minotaur" -authorization OFF -accounting
```

To remove an existing RADIUS action, at the command prompt, type the following command:

rm authentication radiusAction <name>

Example

```
> rm authentication tacacsaction Authn-Act-1  Done
```

Note: In the configuration utility, the term server is used instead of action, but refers to the same task.

1. Navigate to Security > AAA - Application Traffic > Policies > Authentication.
2. In the details pane, on the Servers tab, do one of the following:
   - To create a new authentication server, click Add.
   - To modify an existing authentication server, select the server, and then click Open.
3. In the Create Authentication Server or Configure Authentication Server dialog box, type or select values for the parameters.
   - Name*—radiusActionName (Cannot be changed for a previously configured action)
   - Authentication Type*—authtype (Set to RADIUS, cannot be changed)
   - IP Address*—serverip <IP>
   - IPV6*—Select the checkbox if the server IP is an IPv6 IP. (No command line equivalent.)
   - Port*—serverPort
   - Time-out (seconds)*—authTimeout
4. Click Create or OK, and then click Close. The policy that you created appears in the Authentication Policies and Servers page.

At the command prompt, type the following commands in the order shown to create and bind an authentication policy and verify the configuration:

- add authentication negotiatePolicy <name> <rule> <reqAction>
- show authentication localPolicy <name>
- bind authentication vserver <name> -policy <policyname> [-priority <priority>] [-secondary]]
- show authentication vserver <name>

Example

```
> add authentication localPolicy Authn-Pol-1 ns_true  Done  > show authentication localPolicy  1)  Name: Authn-Pol-1  Rule: ns_true  Request action: LOCAL
```

At the command prompt, type the following commands to modify an existing authentication policy:
set authentication localPolicy <name> <rule> [-reqaction <action>]
Example

```
> set authentication localPolicy Authn-Pol-1 'ns_true'  Done
```

At the command prompt, type the following command to remove an authentication policy:
rm authentication localPolicy <name>
Example

```
> rm authentication localPolicy Authn-Pol-1  Done
```

1. Navigate to Security > AAA - Application Traffic > Policies > Authentication, and then select the type of policy that you want to create.
2. In the details pane, on the Policies tab, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the action, and then click Open .
3. In the Create Authentication Policy or Configure Authentication Policy dialog, type or select values for the parameters.
   - Name*—policyname (Cannot be changed for a previously configured action)
   - Authentication Type*—authtype
   - Server*—authVsName
   - Expression*—rule (You enter expressions by first choosing the type of expression in the leftmost drop-down list beneath the Expression window, and then by typing your expression directly into the expression text area, or by clicking Add to open the Add Expression dialog box and using the drop-down lists in it to construct your expression.)
4. Click Create or OK. The policy that you created appears in the Authentication Policies and Servers page.
5. Click the Servers tab, and in the details pane do one of the following:
   - To use an existing server, select it, and then click Open Edit.
   - To create a new server, click Add, and follow the instructions.
6. If you want to designate this policy as a secondary authentication policy, on the Authentication tab, click Secondary. If you want to designate this policy as a primary authentication policy, skip this step.
7. Click Insert Policy.
8. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
9. In the Priority column to the left, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
10. Click OK. A message appears in the status bar, stating that the policy has been configured successfully.

LDAP Authentication Policies

Aug 13, 2014

As with other types of authentication policies, a Lightweight Directory Access Protocol (LDAP) authentication policy is comprised of an expression and an action. After creating an authentication policy, you bind it to an authentication virtual server and assign a priority to it. When binding it, you also designate it as either a primary or a secondary policy. In addition to standard authentication functions, LDAP can search other active directory (AD) servers for user accounts for users that do not exist locally. This function is called referral support or referral chasing.

Normally you configure the NetScaler ADC to use the IP address of the authentication server during authentication. With LDAP authentication servers, you can also configure the ADC to use the FQDN of the LDAP server instead of its IP address to authenticate users. Using an FQDN can simplify an otherwise much more complex AAA configuration in environments where the authentication server might be at any of several IP addresses, but always uses a single FQDN. To configure authentication by using a server's FQDN instead of its IP address, you follow the normal configuration process except when creating the authentication action. When creating the action, you use the **serverName** parameter instead of the **serverIP** parameter, and substitute the server's FQDN for its IP address.

Before you decide whether to configure the ADC to use the IP or the FQDN of your LDAP server to authenticate users, consider that configuring AAA to authenticate to an FQDN instead of an IP address adds an extra step to the authentication process. Each time the ADC authenticates a user, it must resolve the FQDN. If a great many users attempt to authenticate simultaneously, the resulting DNS lookups might slow the authentication process.

LDAP referral support is disabled by default and cannot be enabled globally. It must be explicitly enabled for each LDAP action. You must also make sure that the AD server accepts the same **binddn** credentials that are used with the referring (GC) server. To enable referral support, you configure an LDAP action to follow referrals, and specify the maximum number of referrals to follow.

If referral support is enabled, and the NetScaler ADC receives an LDAP_REFERRAL response to a request, AAA follows the referral to the active directory (AD) server contained in the referral and performs the update on that server. First, AAA looks up the referral server in DNS, and connects to that server. If the referral policy requires SSL/TLS, it connects via SSL/TLS. It then binds to the new server with the **binddn** credentials that it used with the previous server, and performs the operation which generated the referral. This feature is transparent to the user.

Note: These instructions assume that you are already familiar with the LDAP protocol and have already configured your chosen LDAP authentication server.
For more information about setting up authentication policies in general, see "Authentication Policies". For more information about NetScaler expressions, which are used in the policy rule, see the *Citrix NetScaler Policy Configuration and Reference Guide* at "Policies and Expressions."

At the command prompt, type the following commands:

- set authentication ldapAction <name> -followReferrals ON
- set authentication ldapAction <name> -maxLDAPReferrals <integer>

Example

```
> set authentication ldapAction ldapAction-1 -followReferrals ON
```

```
set authentication ldapAction ldapAction-1 -maxLDAPReferrals 2
```

Note: In the configuration utility, the term server is used instead of action, but refers to the same task.

1. Navigate to Security > AAA - Application Traffic > Policies > LDAP.
2. In the details pane, on the Servers tab, select the LDAP server that you want to configure, and then click Edit.
3. In the Configure Authentication Server dialog, scroll down to the Referrals check box, and select it.
4. In the Maximum Referral Level text box, type the maximum number of referrals to allow.
5. Click OK, and then click Close.

RADIUS Authentication Policies

Aug 21, 2014

As with other types of authentication policies, a Remote Authentication Dial In User Service (RADIUS) authentication policy is comprised of an expression and an action. After creating an authentication policy, you bind it to an authentication virtual server and assign a priority to it. When binding it, you also designate it as either a primary or a secondary policy. However, setting up a RADIUS authentication policy has certain special requirements that are described below.

Normally you configure the NetScaler ADC to use the IP address of the authentication server during authentication. With RADIUS authentication servers, you can now configure the ADC to use the FQDN of the RADIUS server instead of its IP address to authenticate users. Using an FQDN can simplify an otherwise much more complex AAA configuration in environments where the authentication server might be at any of several IP addresses, but always uses a single FQDN. To configure authentication by using a server's FQDN instead of its IP address, you follow the normal configuration process except when creating the authentication action. When creating the action, you substitute the **serverName** parameter for the **serverIP** parameter.

Before you decide whether to configure the ADC to use the IP or the FQDN of your RADIUS server to authenticate users, consider that configuring AAA to authenticate to an FQDN instead of an IP address adds an extra step to the authentication process. Each time the ADC authenticates a user, it must resolve the FQDN. If a great many users attempt to authenticate simultaneously, the resulting DNS lookups might slow the authentication process.

Note: These instructions assume that you are already familiar with the RADIUS protocol and have already configured your chosen RADIUS authentication server.
For more information about setting up authentication policies in general, see "Authentication Policies." For more information about NetScaler expressions, which are used in the policy rule, see the *Citrix NetScaler Policy Configuration and Reference Guide* at "Policies and Expressions."

If you authenticate to a RADIUS server, you need to add an explicit authentication action. To do this, at the command prompt, type the following command:

add authentication radiusAction <name> [-serverip <IP> | -serverName] <FQDN>] [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip (ENABLED | DISABLED)] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting (ON | OFF)] [-pwdVendorID <positive_integer> [-pwdAttributeType <positive_integer>]] [-defaultAuthenticationGroup <string>] [-callingstationid (ENABLED | DISABLED)]

Example

The following example adds a RADIUS authentication action named **Authn-Act-1**, with the server IP **10.218.24.65**, the server port **1812**, the authentication timeout **15** minutes, the radius key **WareTheLorax**, NAS IP disabled, and NAS ID **NAS1**.

```
> add authentication radiusaction Authn-Act-1 -serverip 10.218.24.65 -serverport 1812
        -authtimeout 15 -radkey WareTheLorax -radNASip DISABLED -radNASid NAS1
Done
```

The following example adds the same RADIUS authentication action, but using the server FQDN **rad01.example.com** instead of the IP.

```
> add authentication radiusaction Authn-Act-1 -serverName rad01.example.com -serverport 1812
        -authtimeout 15 -radkey WareTheLorax -radNASip DISABLED -radNASid NAS1
Done
```

To configure an existing RADIUS action, at the NetScaler command prompt, type the following command:

set authentication radiusAction <name> [-serverip <IP> | -serverName] <FQDN>] [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip (ENABLED | DISABLED)] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting (ON | OFF)] [-pwdVendorID <positive_integer> [-pwdAttributeType <positive_integer>]] [-defaultAuthenticationGroup <string>] [-callingstationid (ENABLED | DISABLED)]

To remove an existing RADIUS action, at the command prompt, type the following command:

rm authentication radiusAction <name>

Example

```
> rm authentication radiusaction Authn-Act-1
Done
```

Note: In the configuration utility, the term server is used instead of action, but refers to the same task.

1. Navigate to Security > AAA - Application Traffic > Policies > Authentication > Radius.
2. In the details pane, on the Servers tab, do one of the following:
   - To create a new RADIUS server, click Add.
   - To modify an existing RADIUS server, select the server, and then click Edit.
3. In the Create Authentication RADIUS Server or Configure Authentication RADIUS Server dialog, type or select values for the parameters. To fill out parameters that appear beneath Send Calling Station ID, expand Details.
   - Name*—radiusActionName (Cannot be changed for a previously configured action)
   - Authentication Type*—authtype (Set to RADIUS, cannot be changed)
   - Server Name / IP Address*—Choose either Server Name or Server IP
     - Server Name*—serverName <FQDN>
     - IP Address*—serverIp <IP> If the server is assigned an IPv6 IP address, select the IPv6 check box.
   - Port*—serverPort
   - Time-out (seconds)*—authTimeout
   - Secret Key*—radKey (RADIUS shared secret.)
   - Confirm Secret Key*—Type the RADIUS shared secret a second time. (No command line equivalent.)
   - Send Calling Station ID—callingstationid
   - Group Vendor Identifier—radVendorID
   - Group Attribute Type—radAttributeType
   - IP Address Vendor Identifier—ipVendorID
   - pwdVendorID—pwdVendorID
   - Password Encoding—passEncoding
   - Default Authentication Group—defaultAuthenticationGroup

- NAS ID—radNASid
- Enable NAS IP address extraction—radNASip
- Group Prefix—radGroupsPrefix
- Group Separator—radGroupSeparator
- IP Address Attribute Type—ipAttributeType
- Password Attribute Type—pwdAttributeType
- Accounting—accounting

4. Click Create or OK. The policy that you created appears in the Authentication Policies and Servers page.

SAML Authentication Policies

Sep 09, 2014

The Security Assertion Markup Language (SAML) is an XML-based standard for exchanging authentication and authorization tokens between servers which authenticate users (the *Identity Provider* or *IdP*) and servers that host user applications (*Service Providers*). The NetScaler ADC supports SAML authentication and authorization with HTTP POST-binding, in which the ADC responds to user requests with a 200 OK that contains a form-auto post with the required authentication token.

The NetScaler ADC supports attribute extraction from SAML assertions, and encrypted SAML assertions. The NetScaler implementation of SAML allows signing certificates of less than 2048 bits, but displays a warning message. It also supports the SHA256 hash algorithm for signatures and digests. Citrix recommends that all signing certificates be of at least 2048 bits, and that you use SHA256 as SHA-1 is no longer considered secure.

As with other types of NetScaler authentication policies, a SAML authentication policy is comprised of an expression and an action. After creating an authentication policy, you bind it to an authentication virtual server and assign a priority to it. When binding it, you also designate it as either a primary or a secondary policy. However, setting up a SAML authentication policy has certain special requirements that are described below.

Note: These instructions assume that you are already familiar with the SAML protocol and have already configured your chosen SAML authentication server.
For more information about setting up authentication policies in general, see "Authentication Policies". For more information about NetScaler expressions, which are used in the policy rule, see the *Citrix NetScaler Policy Configuration and Reference Guide* at ""Policies and Expressions"."

If you authenticate to a SAML server, you need to add an explicit authentication action. To do this, at the command prompt, type the following command:

add authentication samlaction <name> [-samlIdPCertName <string>] [-samlSigningCertName <string>] [-samlRedirectUrl <string>] [-samlACSIndex <positive_integer>] [-samlUserField <string>] [-samlRejectUnsignedAssertion (ON | OFF)] [-samlIssuerName <string>] [-samlTwoFactor (ON | OFF)] [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>] [-signatureAlg (RSA-SHA1 | RSA-SHA256)] [-digestMethod (SHA1 | SHA256)]

Example

The following example adds a SAML authentication action named Authn-Act-1.

```
> add authentication samlaction Authn-Act-1 -samlIdPCertName samlcert1
        -samlSigningCertName ssigncert1 -samlRedirectUrl https://login.example.com/logon_fail.html
        -samlUserField userfield1 -samlRejectUnsignedAssertion ON -samlIssuerName Issuer
        -samlTwoFactor ON -defaultAuthenticationGroup group -signatureAlg RSA-SHA256
        -digestMethod SHA256
Done
```

To configure an existing SAML action, at the command prompt, type the following command:

set authentication samlaction <name> [-samlIdPCertName <string>] [-samlSigningCertName <string>] [-samlRedirectUrl <string>] [-samlUserField <string>] [-samlRejectUnsignedAssertion (ON | OFF)] [-samlIssuerName <string>] [-samlTwoFactor (ON | OFF)] [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>] [-signatureAlg (RSA-SHA1 | RSA-SHA256)] [-digestMethod (SHA1 | SHA256)]

Example

```
> add authentication samlaction Authn-Act-1 -samlIdPCertName samlcert1
        -samlSigningCertName ssigncert1 -samlRedirectUrl https://login.example.com/logon_fail.html
        -samlUserField userfield1 -samlRejectUnsignedAssertion ON -samlIssuerName Issuer
        -samlTwoFactor ON -defaultAuthenticationGroup group -signatureAlg RSA-SHA256
        -digestMethod SHA256
Done
```

To remove an existing SAML action, at the command prompt, type the following command:

rm authentication samlaction <name>

Example

```
> rm authentication samlaction Authn-Act-1
Done
```

Note: In the configuration utility, the term server is used instead of action, but refers to the same task.

1. Navigate to AAA - Application Traffic > Policies > Authentication > SAML.
2. In the details pane, on the Servers tab, do one of the following:
   - To create a new SAML server, click Add.
   - To modify an existing SAML server, select the server, and then click Open.
3. In the Create Authentication Server or Configure Authentication Server dialog box, type or select values for the parameters:
   - Name*—policyname (Cannot be changed for a previously configured action)
   - Authentication Type*—authtype (Set to SAML, cannot be changed)
   - IDP Certificate Name*—samlIDPcertname
   - Redirect URL*—samlRedirectUrl
   - User Field—samlUserField
   - Signing Certificate Name—samlSigningcertname
   - Issuer Name—samlIssuerName
   - Default Authentication Group—defaultAuthenticationGroup
   - Two Factor—samlTwoFactor
   - Reject Unsigned Assertion—samlRejectUnsignedAssertion
   - ACS Index—samlACSIndex
   - Attribute 1-Attribute 16—attribute1-attribute16 (Used to extract attributes from the SAML assertion.)
4. Click Create or OK. The policy that you created appears in the Authentication Policies and Servers page.

Authorization Policies

Aug 19, 2014

After you create authentication policies, you next create any authorization policies you need. Authorization policies, like other policies, consist of an expression and action. There are only two actions for authorization policies: ALLOW and DENY. ALLOW permits users to access the specified resource; DENY blocks access. The default setting for authorization when no specific policy exists is to deny access to network resources. This means that a user or group can access a particular resource only if an authorization policy explicitly allows access. For optimum security, the best practice is not to change the default setting and to create specific authorization policies for users who need access to specific resources.

Authorization use both default syntax expressions and classic expressions. These expressions are described in detail in the *Citrix NetScaler Policy Configuration and Reference Guide* at "Policies and Expressions."

After you create an authorization policy, you bind it to the appropriate user accounts or groups to put it into effect.

At the NetScaler command prompt, type the following commands to create an authorization policy and verify the configuration:

- add authorization policy <name> <rule> <action>
- show authorization policy <name>

Example

```
> add authorization policy authz-pol-1 "HTTP.REQ.URL.SUFFIX.EQ(\"gif\")" DENY
 Done
> show authorization policy authz-pol-1
1)     Name: authz-pol-1    Rule: HTTP.REQ.URL.SUFFIX.EQ("gif")
       Action: DENY
 Done
>
```

At the command prompt, type the following command to modify an authorization policy:
set authorization policy <name> [-rule <expression>] -action <action>

Example

```
> set authorization policy authz-pol-1 -rule "HTTP.REQ.URL.SUFFIX.EQ(\"gif\")" -action ALLOW
 Done
> show authorization policy authz-pol-1
1)     Name: authz-pol-1    Rule: HTTP.REQ.URL.SUFFIX.EQ("gif")
       Action: ALLOW
 Done
>
```

At the command prompt, type one of the following commands to bind an authorization policy to a user account or group and verify the configuration:

- bind aaa user <userName> [-policy <policyname> [-priority <priority]] [-intranetApplication <appname>] [-urlName

<urlname>] [-intranetIP <intranetip> [<netmask>]]
- show aaa user <userName>
- bind aaa group <groupName> [-policy <policyname> [-priority <priority]] [-intranetApplication <appname>] [-urlName <urlname>] [-intranetIP <intranetip> [<netmask>]]
- show aaa group <name>

Example

```
> bind aaa user user-hr-1 -policy authz-pol-1
 Done
> show aaa user user-hr-1
      UserName: user-hr-1

            Policy: authz-pol-1, Priority: 0
 Done
> bind aaa group group-1 -policy authz-pol-1
 Done
> show aaa group group-1
      GroupName: group-1

            UserName: user-2
            UserName: user-1

            Policy: authz-pol-1, Priority: 0
 Done
```

At the command prompt, type one of the following commands to unbind an authorization policy from a user account or group:
- unbind aaa user <userName> -policy <policyname>
- unbind aaa group <groupName> -policy <policyname>

Example

```
> unbind aaa user aaa-user-1 -policy auth-pol-1
Done
```

First unbind the policy from all user accounts and groups, and then, at the NetScaler command prompt, type the following command to remove an authorization policy:
rm authorization policy <name>

1. Navigate to Security > AAA - Application Traffic > Authorization.
2. In the details pane, do one of the following:
   - To create a new authorization policy, click Add.
   - To modify an existing authorization policy, select the policy, and then click Open.
3. In the Create Authorization Policy or Configure Authorization Policy dialog, type or select values for the parameters.
   - Name*—policyname(Cannot be changed for a previously configured policy.)

- Action*— action
- Expression*—rule (By default, the Expression box accepts default syntax policies. To switch to the classic syntax view, click Switch to Classic Syntax.)

4. Click Create or OK. The policy that you created appears on the Authorization Policies page.
5. To bind an authorization policy to a user account or group, in the navigation pane, under AAA - Application Traffic, click either Users or Groups, as appropriate, and then add that policy to the user account list:
   1. In the details pane, select the appropriate user account, and then click Open.
   2. Click the Authorization tab.
   3. Click Insert Policy.
   4. Select the policy you want to bind to the user account or group.
   5. In the Priority column, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   6. Click OK.

   A message appears in the status bar, stating that the policy has been configured successfully.

Auditing Policies

Aug 13, 2014

After you create authentication policies, you next create any auditing policies you need. The NetScaler ADC allows auditing of all states and status information, so you can see the event history for any user in chronological order. When you configure auditing on the ADC, you can choose to store the log files locally on the ADC or to send them to a syslog server.

To put your auditing policies into effect, you bind them globally, to a specific authentication virtual server, or to specific user accounts or groups.

At the command prompt, type the following commands to create an auditing policy and verify the configuration:

- add audit nslogPolicy <name> [-rule <rule>] [-action <action>]
- show audit nslogPolicy

Example

```
> add audit nslogPolicy audit-1 ns_true audit_server
 Done
> show audit nslogPolicy
1)     Name: audit-pol Rule: ns_true
       Action: audit_server
2)     Name: audit-1   Rule: ns_true
       Action: audit_server
 Done
```

At the command prompt, type the following commands to modify an auditing policy and verify the configuration:

- set audit nslogPolicy <name> [-rule <expression>] [-action <string>]
- show audit nslogPolicy

Example

```
> set audit nslogPolicy audit-1 ns_true audit_server
 Done
> show audit nslogPolicy
1)     Name: audit-pol Rule: ns_true
       Action: audit_server
2)     Name: audit-1   Rule: ns_true
       Action: audit_server
 Done
```

At the command prompt, type the following commands to globally bind an auditing policy:
bind tm global [-policyName <string> [-priority <positive_integer>]]

Example

```
> bind tm global -policyName Audit-Pol-1 -priority 1000
Done
```

At the command prompt, type the following commands to bind an auditing policy to an authentication virtual server and verify the configuration:

- bind authentication vserver <name> [-policy <string> [-priority <positive_integer>] [-secondary] [-groupExtraction]]
- show authentication vserver [<name>]

Example

```
> bind authentication Vserver Auth-Vserver-2 -policy Authn-Pol-1
 Done
> show authentication Vserver Auth-Vserver-2
      Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
      State: UP
      Client Idle Timeout: 180 sec
      Down state flush: DISABLED
      Disable Primary Vserver On Down : DISABLED
      Authentication : ON
      Current AAA Users: 0
      Authentication Domain: myCompany.employee.com

1)          Primary authentication policy name: Authn-Pol-1 Priority: 0
 Done
```

At the command prompt, type one of the following commands to bind an auditing policy to a user account or a group:

- bind audit <logtype> user <userName> -policy <policyname> [-priority <priority]
- bind audit <logtype> user <userName> -policy <policyname> [-priority <priority]

Example

```
> bind audit nslogPolicy user aaa-user-1 -policyName Audit-Pol-1 -priority 1000
Done
```

At the command prompt, type the following commands to unbind a globally-bound auditing policy:
unbind audit <logtype> global -policy <policyname>
Example

```
> unbind audit nslogPolicy global -policy Audit-Pol-1
Done
```

At the command prompt, type the following commands to unbind an auditing policy from an authentication virtual server:
unbind authentication vserver <name> [-policy <string> [-secondary][-groupExtraction]]
Example


```
> unbind authentication vserver auth-vserver-1 -policyName Audit-Pol-1
Done
```

At the command prompt, type one of the following commands to unbind an auditing policy from a user account or a group:

- unbind audit <logtype> user <userName> -policy <policyname>
- unbind audit <logtype> group <groupName> -policy <policyname>

Example

```
> unbind audit nslogPolicy group aaa-group-1 -policyName Audit-Pol-1
Done
```

First unbind the policy from all users and groups, and then, at the command prompt, type the following command to remove an auditing policy:
rm audit <logtype> <policyname>

1. Navigate to Security > AAA - Application Traffic > Policies > Auditing.
2. Choose the type of auditing policy that you want to create.
   - To create a policy that logs to syslog, expand Syslog.
   - To create a policy that logs to nslog, expand Nslog.

   Note: The dialogs for the two types of policies are nearly identical.
3. In the details pane, do one of the following:
   - To create a new auditing policy, click Add.
   - To modify an existing auditing policy, select the policy, and then click Edit.
4. In the Create Audit Policy or Configure Audit Policy dialog, type or select values for the parameters.
   - Name*—policyname (Cannot be changed for a previously configured policy.)
   - Auditing type*—logtype (When creating auditing policies by using the configuration utility, you cannot specify a rule.)
   - Server*—action
5. Click Create or OK. The policy that you created appears in the Auditing Policies page.
6. Click OK.
7. To globally bind an auditing policy, in the details pane, click Global Bindings and fill in the Bind/Unbind Audit Policies to Global dialog box.
   1. Select the name of the audit policy you want to globally bind.
   2. Click OK.

   A message appears in the status bar, stating that the policy has been configured successfully.
8. To bind an auditing policy to an authentication virtual server, in the navigation pane, click Virtual Servers, and add that policy to the authentication policies list.
   1. In the details pane, select the appropriate virtual server, and then click Open.
   2. Click the Policies tab.
   3. Click Insert Policy.
   4. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
   5. In the Priority column, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   6. Click OK.
9. To bind an auditing policy to a user account or group, in the navigation pane, click Users or Groups, and add that policy to the user account list.

1. In the details pane, select the appropriate user account, and then click Open.
2. Click the Policies tab.
3. Click Insert Policy.
4. Choose the policy you want to bind to the group from the drop-down list.
5. In the Priority column, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
6. Click OK.

A message appears in the status bar, stating that the policy has been configured successfully.

Session Settings

Apr 18, 2013

After you configure your authentication, authorization, and auditing profiles, you configure session settings to customize your user sessions. The session settings are:

The session timeout.

Controls the period after which the user is automatically disconnected and must authenticate again to access your intranet.

The default authorization setting.

Determines whether the NetScaler appliance will by default allow or deny access to content for which there is no specific authorization policy.

The single sign-on setting.

Determines whether the NetScaler appliance will log users onto all web applications automatically after they authenticate, or will pass users to the web application logon page to authenticate for each application.

The credential index setting.

Determines whether the NetScaler appliance will use primary or the secondary authentication credentials for single signon.

To configure the session settings, you can take one of two approaches. If you want different settings for different user accounts or groups, you create a profile for each user account or group for which you want to configure custom sessions settings. You also create policies to select the connections to which to apply particular profiles, and you bind the policies to users or groups. You can also bind a policy to the authentication virtual server that handles the traffic to which you want to apply the profile.

If you want the same settings for all sessions, or if you want to customize the default settings for sessions that do not have specific profiles and policies configured, you can simply configure the global session settings.

Session Profiles

Aug 19, 2014

To customize your user sessions, you first create a session profile. The session profile allows you to override global settings for any of the session parameters.

Note: The terms “session profile” and “session action” mean the same thing.

At the command prompt, type the following commands to create a session profile and verify the configuration:

- add tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ENABLED | DISABLED)] [-persistentCookieValidity <minutes>]
- show tm sessionAction <name>

Example

```
> add tm sessionAction session-profile -sessTimeout 30 -defaultAuthorization ALLOW
 Done
> show tm sessionAction session-profile
1)     Name: session-profile
       Authorization action : ALLOW
       Session timeout: 30 minutes
 Done
```

At the command prompt, type the following commands to modify a session profile and verify the configuration:

- set tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ENABLED | DISABLED)] [-persistentCookieValidity <minutes>]
- show tm sessionAction

Example

```
> set tm sessionAction session-profile -sessTimeout 30 -defaultAuthorization ALLOW
 Done
> show tm sessionAction session-profile
1)     Name: session-profile
       Authorization action : ALLOW
       Session timeout: 30 minutes
 Done
```

At the command prompt, type the following command to remove a session profile:

rm tm sessionAction <name>

1. Navigate to Security > AAA - Application Traffic > Session.
2. In the details pane, click the Profiles tab.

3. On the Profiles tab, do one of the following:
   - To create a new session profile, click Add.
4. In the Create TM Session Profile or Configure TM Session Profile dialog, type or select values for the parameters.
   - Name*—actionname (Cannot be changed for a previously configured session action.)
   - Session Time-out—sesstimeout
   - Default Authorization Action—defaultAuthorizationAction
   - Single Signon to Web Applications—sso
   - Credential Index—ssocredential
   - Single Sign-on Domain—ssoDomain
   - HTTPOnly Cookie—httpOnlyCookie
   - Enable Persistent Cookie—persistentCookie
   - Persistent Cookie Validity—persistentCookieValidity
5. Click Create or OK. The session profile that you created appears in the Session Policies and Profiles pane.

Session Policies

Aug 19, 2014

After you create one or more session profiles, you create session policies and then bind the policies globally or to an authentication virtual server to put them into effect.

At the command prompt, type the following commands to create a session policy and verify the configuration:

- add tm sessionPolicy <name> <rule> <action>
- show tm sessionPolicy <name>

Example

```
> add tm sessionPolicy session-pol "URL == /*.gif" session-profile
 Done
> show tm sessionPolicy session-pol
1)     Name: session-pol       Rule: URL == '/*.gif'
       Action: session-profile
 Done
```

At the command prompt, type the following commands to modify a session policy and verify the configuration:

- set tm sessionPolicy <name> [-rule <expression>] [-action <action>]
- show tm sessionPolicy <name>

Example

```
> set tm sessionPolicy session-pol "URL == /*.gif" session-profile
 Done
> show tm sessionPolicy session-pol
1)     Name: session-pol       Rule: URL == '/*.gif'
       Action: session-profile
 Done
```

At the command prompt, type the following commands to globally bind a session policy and verify the configuration:
bind tm global -policyName <policyname> [-priority <priority>]

Example

```
> bind tm global -policyName session-pol
 Done

> show tm sessionPolicy session-pol
1)     Name: session-pol       Rule: URL == '/*.gif'
       Action: session-profile
       Policy is bound to following entities
       1) TM GLOBAL    PRIORITY : 0
 Done
```

At the command prompt, type the following command to bind a session policy to an authentication virtual and verify the configuration:
bind authentication vserver <name> -policy <policyname> [-priority <priority>]
Example

```
> bind authentication vserver auth-vserver-1 -policyName Session-Pol-1 -priority 1000
Done
```

At the command prompt, type the following commands to unbind a session policy from an authentication virtual server and verify the configuration:
unbind authentication vserver <name> -policy <policyname>
Example

```
> unbind authentication vserver auth-vserver-1 -policyName Session-Pol-1
Done
```

At the command prompt, type the following commands to unbind a globally-bound session policy:
unbind tm global -policyName <policyname>
Example

```
> unbind tm global -policyName Session-Pol-1
Done
```

First unbind the session policy from global, and then, at the command prompt, type the following commands to remove a session policy and verify the configuration:
rm tm sessionPolicy <name>
Example

```
> rm tm sessionPolicy Session-Pol-1
Done
```

1. Navigate to Security > AAA - Application Traffic > Session.
2. In the details pane, on the Policies tab, do one of the following:
   - To create a new session policy, click Add.
   - To modify an existing session policy, select the policy, and then click Open.
3. In the Create Session Policy or Configure Session Policy dialog, type or select values for the parameters.
   - Name*—policyname (Cannot be changed for a previously configured session policy.)
   - Request Profile*—actionname
   - Expression*—rule (You enter expressions by first choosing the type of expression in the leftmost drop-down list beneath the Expression text area and then typing your expression directly into the expression text area, or by clicking Add to open the Add Expression dialog box and using the drop-down lists in it to construct your expression.)

4. Click Create or OK. The policy that you created appears in the details pane of the Session Policies and Profiles page.
5. To globally bind a session policy, in the details pane, click Global Bindings, and fill in the dialog.
    1. Select the name of the session policy you want to globally bind.
    2. Click OK.
6. To bind a session policy to an authentication virtual server, in the navigation pane, click Virtual Servers, and add that policy to the policies list.
    1. In the details pane, select the virtual server, and then click Open.
    2. Click the Policies tab.
    3. Click Insert Policy.
    4. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
    5. In the Priority column to the left, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
    6. Click OK.

    A message appears in the status bar, stating that the policy has been configured successfully.

Global Session Settings

Aug 13, 2014

In addition to or instead of creating session profiles and policies, you can configure global session settings. These settings control the session configuration when there is no explicit policy overriding them.

At the command prompt, type the following commands to configure the global session settings and verify the configuration:

set tm sessionParameter [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ENABLED | DISABLED)] [-persistentCookieValidity <minutes>]

Example

```
> set tm sessionParameter -sessTimeout 30
  Done
> set tm sessionParameter -defaultAuthorizationAction DENY
  Done
> set tm sessionParameter -SSO ON
  Done
> set tm sessionParameter -ssoCredential PRIMARY
  Done
```

1. Navigate to Security > AAA - Application Traffic
2. In the details pane, under Settings, click Change global settings.
3. In the Global Session Settings dialog, type or select values for the parameters.
   - Session Time-out—sessTimeout
   - Default Authorization Action—defaultAuthorizationAction
   - Single Sign-on to Web Applications—sso
   - Credential Index—ssoCredential
   - Single Sign-on Domain—ssoDomain
   - HTTPOnly Cookie—httpOnlyCookie
   - Enable Persistent Cookie—persistentCookie
   - Persistent Cookie Validity (minutes)—persistentCookieValidity
   - Home Page—homepage
4. Click OK.

Traffic Settings

Sep 24, 2013

If you use forms-based or SAML single sign-on (SSO) for your protected applications, you configure that feature in the Traffic settings. SSO enables your users to log on once to access all protected applications, rather than requiring them to log on separately to access each one.

Forms-based SSO allows you to use a web form of your own design as the sign-on method instead of a generic pop-up window. You can therefore put you company logo and other information you might want your users to see on the logon form. SAML SSO allows you to configure one NetScaler appliance or virtual appliance instance to authenticate to another NetScaler appliance on behalf of users who have authenticated with the first appliance.

To configure either type of SSO, you first create a forms or SAML SSO profile. Next, you create a traffic profile and link it to the SSO profile you created. Next, you create a policy, link it to the traffic profile. Finally, you bind the policy globally or to an authentication virtual server to put your configuration into effect.

Traffic Profiles

Aug 19, 2014

After creating at least one forms or SAML sso profile, you must next create a traffic profile.

Note: In this feature, the terms “profile” and “action” mean the same thing.

At the command prompt, type:

add tm trafficAction <name> [-appTimeout <mins>] [-SSO (ON | OFF) [-formSSOAction <string>]] [-persistentCookie (ENABLED | DISABLED)] [-InitiateLogout (ON | OFF)]

Example

add tm trafficAction Traffic-Prof-1 –appTimeout 10 -SSO ON -formSSOAction SSO-Prof-1

At the command prompt, type:

set tm trafficAction <name> [-appTimeout <mins>] [-SSO (ON | OFF) [-formSSOAction <string>]] [-persistentCookie (ENABLED | DISABLED)] [-InitiateLogout (ON | OFF)]

Example

set tm trafficAction Traffic-Prof-1 –appTimeout 10 -SSO ON -formSSOAction SSO-Prof-1

At the command prompt, type:

rm tm trafficAction <name>

Example

rm tm trafficAction Traffic-Prof-1

1. Navigate to Security > AAA - Application Traffic > Traffic.
2. In the details pane, click the Profiles tab.
3. On the Profiles tab, do one of the following:
   - To create a new traffic profile, click Add.
   - To modify an existing traffic profile, select the profile, and then click Open.
4. In the Create Traffic Profile or Configure Traffic Profile dialog box, specify values for the parameters.
   - Name*—name (Cannot be changed for a previously configured session action.)
   - AppTimeout—appTimeout
   - Single Sign-On—SSO
   - Form SSO Action—formSSOAction
   - SAML SSO Action—samlSSOAction
   - Enable Persistent Cookie—persistentCookie
   - Initiate Logout—InitiateLogout
5. Click Create or OK. The traffic profile that you created appears in the Traffic Policies, Profiles, and either the Form SSO Profiles or SAML SSO Profiles pane, as appropriate.

Traffic Policies

Aug 19, 2014

After you create one or more form SSO and traffic profiles, you create traffic policies and then bind the policies, either globally or to a traffic management virtual server, to put them into effect.

At the command prompt, type:

add tm trafficPolicy <name> <rule> <action>

Example

add tm trafficPolicy Traffic-Pol-1 "HTTP.REQ.HEADER("Cookie").CONTAINS("login=true")" Traffic-Prof-1

At the command prompt, type:

set tm trafficPolicy <name> <rule> <action>

Example

set tm trafficPolicy Traffic-Pol-1 "HTTP.REQ.HEADER("Cookie").CONTAINS("login=true")" Traffic-Prof-1

At the command prompt, type:

bind tm global -policyName <string> [-priority <priority>]

Example

bind tm global -policyName Traffic-Pol-1

At the command prompt, type one of the following commands:

- bind lb vserver <name> -policy <policyName> [-priority <priority>]
- bind cs vserver <name> -policy <policyName> [-priority <priority>]

Example

bind authentication vserver auth-vserver-1 -policyName Traffic-Pol-1 -priority 1000

At the command prompt, type:

unbind tm global -policyName <policyname>

Example

unbind tm global -policyName Traffic-Pol-1

At the command prompt, type one of the following commands:

- unbind lb vserver <name> -policy <policyname>
- unbind cs vserver <name> -policy <policyname>

Example

```
unbind authentication vserver auth-vserver-1 -policyName Traffic-Pol-1
```

First unbind the session policy from global, and then, at the command prompt, type:

rm tm trafficPolicy <name>

Example

```
rm tm trafficPolicy Traffic-Pol-1
```

1. Navigate to Security > AAA - Application Traffic > Traffic.
2. In the details pane, do one of the following:
   - To create a new session policy, click Add.
   - To modify an existing session policy, select the policy, and then click Open.
3. In the Create Traffic Policy or Configure Traffic Policy dialog, specify values for the parameters.
   - Name*—policyName (Cannot be changed for a previously configured session policy.)
   - Profile*—actionName
   - Expression—rule (You enter expressions by first choosing the type of expression in the leftmost drop-down list beneath the Expression text area and then typing your expression directly into the expression text area, or by clicking Add to open the Add Expression dialog box and using the drop-down lists in it to construct your expression.)
4. Click Create or OK. The policy that you created appears in the details pane of the Session Policies and Profiles page.
5. To globally bind a traffic policy, in the details pane, click Global Bindings and fill in the Bind/Unbind Policies to Global dialog box.
   1. Select the name of the traffic policy you want to globally bind.
   2. Click OK.
6. To bind a traffic policy to an authentication virtual server, in the navigation pane, click Virtual Servers, and add that policy to the policies list.
   1. In the details pane, select the appropriate virtual server, and then click Open.
   2. Click the Policies tab
   3. Click Insert Policy.
   4. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
   5. In the Priority column to the left, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   6. Click OK.

Form SSO Profiles

Jun 08, 2015

To enable and configure forms-based SSO, you first create an SSO profile.

Note:

- Forms-based single sign-on does not work if the form is customized to include Javascript.
- In this feature, the terms “profile” and “action” mean the same thing.

At the command prompt, type:

- add tm formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule <expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)
- show tm formSSOAction [<name>]

Example

```
add tm formSSOAction SSO-Prof-1 -actionURL "/logon.php"
-userField "loginID" -passwdField "passwd"
-nameValuePair "loginID passwd" -responsesize "9096"
-ssoSuccessRule "HTTP.RES.HEADER("Set-Cookie").CONTAINS("LogonID")"
-nvtype STATIC -submitMethod GET
–sessTimeout 10 -defaultAuthorizationAction ALLOW
```

At the command prompt, type:

set tm formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule <expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)

Example

```
set tm formSSOAction SSO-Prof-1 -actionURL "/logon.php"
-userField "loginID" -passwdField "passwd"
-ssoSuccessRule "HTTP.RES.HEADER("Set-Cookie").CONTAINS("LogonID")"
-nameValuePair "loginID passwd" -responsesize "9096"
-nvtype STATIC -submitMethod GET
–sessTimeout 10 -defaultAuthorizationAction ALLOW
```

At the command prompt, type:

rm tm formSSOAction <name>

Example

```
rm tm sessionAction SSO-Prof-1
```

1. Navigate to Security > AAA - Application Traffic > Policies > Traffic.
2. In the details pane, click the Form SSO Profiles tab.
3. On the Form SSO Profiles tab, do one of the following:
   - To create a new form SSO profile, click Add.
   - To modify an existing form SSO profile, select the profile, and then click Edit.
4. In the Create Form SSO Profile or Configure Form SSO Profile dialog, specify values for the parameters:
   - Name*—name (Cannot be changed for a previously configured session action.)
   - Action URL*—actionURL
   - User Name Field*—userField
   - Password Field*—passField
   - Expression*—ssoSuccessRule
   - Name Value Pair—nameValuePair
   - Response Size—responsesize
   - Extraction—nvtype
   - Submit Method—submitMethod
5. Click Create or OK, and then click Close. The form SSO profile that you created appears in the Traffic Policies, Profiles, and Form SSO Profiles pane.

SAML SSO Profiles

Jun 12, 2014

To enable and configure SAML-based SSO, you first create a SAML SSO profile.

At the command prompt, type:
add tm samlSSOProfile <name> -samlSigningCertName <string> -assertionConsumerServiceURL <URL> -relaystateRule <expression> -sendPassword (ON | OFF) [-samlIssuerName <string>]

Example

```
add tm samlSSOProfile saml-SSO-Prof-1 -samlSigningCertName "Example, Inc."
-assertionConsumerServiceURL "https://service.example.com" -relaystateRule "true"
-sendPassword "ON" -samlIssuerName "Example, Inc."
```

At the command prompt, type:

set tm samlSSOProfile <name> -samlSigningCertName <string> -assertionConsumerServiceURL <URL> -relaystateRule <expression> -sendPassword (ON | OFF) [-samlIssuerName <string>]

Example

```
set tm samlSSOProfile saml-SSO-Prof-1 -samlSigningCertName "Example, Inc."
-assertionConsumerServiceURL "https://service.example.com" -relaystateRule "true"
-sendPassword "ON" -samlIssuerName "Example, Inc."
```

At the command prompt, type:

rm tm samlSSOProfile <name>

Example

```
rm tm sessionAction saml-SSO-Prof-1
```

1. Navigate to Security > AAA - Application Traffic > Policies > Traffic.
2. In the details pane, click the SAML SSO Profiles tab.
3. On the SAML SSO Profiles tab, do one of the following:
   - To create a new SAML SSO profile, click Add.
   - To modify an existing SAML SSO profile, select the profile, and then click Open.
4. In the Create SAML SSO Profiles or the Configure SAML SSO Profiles dialog box, set the following parameters:
   - Name*
   - Signing Certificate Name*
   - ACS URL*
   - Relay State Rule*
   - Send Password
   - Issuer Name

5. Click Create or OK, and then click Close. The SAML SSO profile that you created appears in the Traffic Policies, Profiles, and SAML SSO Profiles pane.

Authenticating with Client Certificates

Aug 13, 2014

Web sites that contain sensitive content, such as online banking websites or websites with employee personal information, sometimes require client certificates for authentication. To configure AAA to authenticate users on the basis of client-side certificate attributes, you first enable client authentication on the traffic management virtual server and bind the root certificate to the authentication virtual server. Then, you implement one of two options. You can configure the default authentication type on the authentication virtual server as CERT, or you can create a certificate action that defines what the NetScaler ADC must do to authenticate users on the basis of a client certificate. In either case, your authentication server must support CRLs. You configure the ADC to extract the user name from the SubjectCN field or another specified field in the client certificate.

When the user tries to log on to an authentication virtual server for which an authentication policy is not configured, and a global cascade is not configured, the user name information is extracted from the specified field of the certificate. If the required field is extracted, the authentication succeeds. If the user does not provide a valid certificate during the SSL handshake, or if the user name extraction fails, authentication fails. After it validates the client certificate, the ADC presents a logon page to the user.

The following procedures assume that you have already created a functioning AAA configuration, and therefore they explain only how to enable authentication by using client certificates. These procedures also assume that you have obtained your root certificate and client certificates and have placed them on the ADC in the /nsconfig/ssl directory.

At the command prompt, type the following commands, in the order shown, to configure the certificate and verify the configuration:

- add ssl certKey <certkeyName> -cert <certFile> -key <keyFile> -password -inform <inform> -expiryMonitor <expiryMonitor> -notificationPeriod <notificationPeriod>
- bind ssl certKey <certkeyName> -vServer <certkeyName> -CA -crlCheck Mandatory
- show ssl certKey [<certkeyName>]
- set aaa parameter -defaultAuthType CERT
- show aaa parameter
- set aaa certParams -userNameField "Subject:CN"
- show aaa certParams

1. Navigate to Security > AAA - Application Traffic > Virtual Servers.
2. In the details pane, select the virtual server that you want to configure to handle client certificate authentication, and then click Edit.
3. On the Configuration page, under Certificates, click the right arrow (>) to open the CA Cert Key installation dialog.
4. In theCA Cert Key dialog box, click Insert.
5. In the CA Cert Key - SSL Certificates dialog box, click Install.
6. In the Install Certificate dialog box, set the following parameters, whose names correspond to the CLI parameter names as shown:
   - Certificate-Key Pair Name*—certkeyName
   - Certificate File Name—certFile
   - Key File Name—keyFile

- Certificate Format—inform
- Password—password
- Certificate Bundle—bundle
- Notify When Expires—expiryMonitor
- Notification Period—notificationPeriod

7. Click Install, and then click Close.
8. In the CA Cert Key dialog box, in the Certificate list, select the root certificate.
9. Click Save.
10. Click Back to return to the main configuration screen.
11. Navigate to Security > AAA - Application Traffic > Policies > Authentication > CERT.
12. In the details pane, select the policy you want to configure to handle client certificate authentication, and then click Edit.
13. In the Configure Authentication CERT Policy dialog, Server drop-down list, select the virtual server you just configured to handle client certificate authentication.
14. Click OK. A message appears in the status bar, stating that the configuration completed successfully.

Configuring AAA with Commonly Used Protocols

Mar 15, 2013

Configuring the NetScaler for Authentication, Authorization, and Auditing (AAA) needs a specific setup on the NetScaler and clients' browsers. The configuration varies with the protocol used for AAA.

For more information about configuring the NetScaler for Kerberos authentication, see Handling Authentication, Authorization and Auditing with Kerberos/NTLM.

Handling Authentication, Authorization and Auditing with Kerberos/NTLM

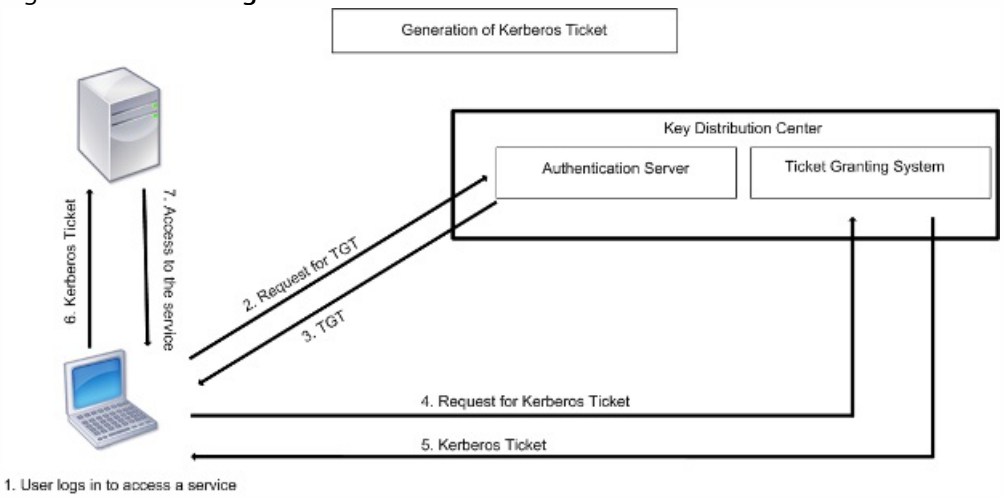
Mar 27, 2012

Kerberos, a computer network authentication protocol, provides secure communication over the Internet. Designed primarily for client-server applications, it provides for mutual authentication by which the client and server can each ensure the other's authenticity. Kerberos uses a trusted third party, referred to as Key Distribution Center (KDC). A KDC consists of an Authentication Server (AS), which authenticates a user, and a Ticket Granting Server (TGS).

Each entity on the network (client or server) has a secret key that is known only to itself and the KDC. The knowledge of this key implies authenticity of the entity. For communication between two entities on the network, the KDC generates a session key, referred to as the Kerberos ticket or service ticket. The client makes a request to the AS for credentials for a specific server. The client then receives a ticket, referred to as Ticket Granting Ticket (TGT). The client then contacts the TGS, using the TGT it received from the AS to prove its identity, and asks for a service. If the client is eligible for the service, the TGS issues a Kerberos ticket to the client. The client then contacts the server hosting the service (referred to as the service server), using the Kerberos ticket to prove that it is authorized to receive the service. The Kerberos ticket has a configurable lifetime. The client authenticates itself with the AS only once. If it contacts the physical server multiple times, it reuses the AS ticket.

The following figure shows the basic functioning of the Kerberos protocol.

Figure 1. **Functioning of Kerberos**



Kerberos authentication has the following advantages:

- Faster authentication. When a physical server gets a Kerberos ticket from a client, the server has enough information to authenticate the client directly. It does not have to contact a domain controller for client authentication, and therefore the authentication process is faster.
- Mutual authentication. When the KDC issues a Kerberos ticket to a client and the client uses the ticket to access a service, only authenticated servers can decrypt the Kerberos ticket. If the virtual server on the NetScaler is able to decrypt the Kerberos ticket, you can conclude that both the virtual server and client are authenticated. Thus, the authentication of the server happens along with the authentication of the client.
- Single sign-on between Windows and other operating systems that support Kerberos.

Kerberos authentication may have the following disadvantages:

- Kerberos has strict time requirements; the clocks of the involved hosts must be synchronized with the Kerberos server clock to ensure that the authentication does not fail. You can mitigate this disadvantage by using the Network Time Protocol daemons to keep the host clocks synchronized. Kerberos tickets have an availability period, which you can configure.
- Kerberos needs the central server to be available continuously. When the Kerberos server is down, no one can log on. You can mitigate this risk by using multiple Kerberos servers and fallback authentication mechanisms.
- Because all the authentication is controlled by a centralized KDC, any compromise in this infrastructure, such as the user's password for a local workstation being stolen, can allow an attacker to impersonate any user. You can mitigate this risk to some extent by using only a desktop machine or laptop that you trust, or by enforcing preauthentication by means of a hardware-token.

To use Kerberos authentication, you must configure it on the NetScaler appliance and on each client.

How NetScaler Implements Kerberos Authentication

Jun 12, 2014

Note: Kerberos/NTLM authentication is supported only in the NetScaler 9.3 nCore release or later, and it can be used only for AAA traffic management (AAA-TM) virtual servers.

NetScaler handles the components involved in Kerberos authentication in the following way:

Key Distribution Center (KDC)

In the Windows 2000 Server or later versions, the Domain Controller and KDC are part of the Windows Server. If the Windows Server is UP and running, it indicates that the Domain Controller and KDC are configured. The KDC is also the Active Directory server.

Note: All Kerberos interactions are validated with the Windows Kerberos Domain Controller.

Authentication Service and Protocol Negotiation

A NetScaler appliance supports Kerberos authentication on the AAA-TM authentication virtual servers. If the Kerberos authentication fails, the NetScaler uses the NTLM authentication.

By default, Windows 2000 Server and later Windows Server versions use Kerberos for AAA. If you create an authentication policy with NEGOTIATE as the authentication type, the NetScaler attempts to use the Kerberos protocol for AAA and if the client's browser fails to receive a Kerberos ticket, the NetScaler uses the NTLM authentication. This process is referred to as negotiation.

The client may fail to receive a Kerberos ticket in any of the following cases:

- Kerberos is not supported on the client.
- Kerberos is not enabled on the client.
- The client is in a domain other than that of the KDC.
- The Access Directory on the KDC is not accessible to the client.

For Kerberos/NTLM authentication, the NetScaler does not use the data that is present locally on the NetScaler appliance.

Authorization

The traffic management virtual server can be a load balancing virtual server or a content switching virtual server.

Auditing

The NetScaler appliance supports auditing of Kerberos authentication with the following audit logging:

- Complete audit trail of the traffic management end-user activity
- SYSLOG and high performance TCP logging
- Complete audit trail of system administrators
- All system events
- Scriptable log format

Supported Environment

Kerberos authentication does not need any specific environment on the NetScaler. The client (browser) must provide support for Kerberos authentication.

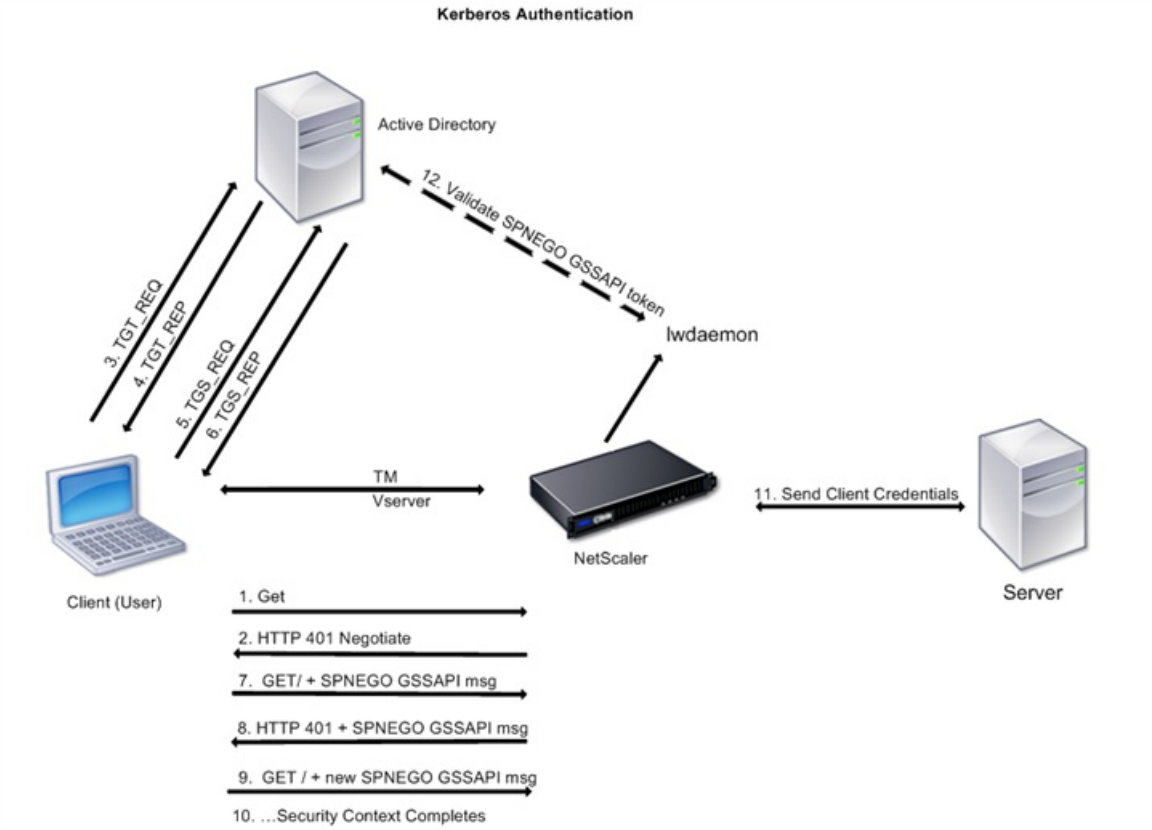
High Availability

In a high availability setup, only the active NetScaler joins the domain. In case of a failover, the NetScaler lwagent daemon joins the secondary NetScaler appliance to the domain. No specific configuration is required for this functionality.

Kerberos Authentication Process

The following figure shows a typical process for Kerberos authentication in the NetScaler environment.

Figure 1. Kerberos Authentication Process on NetScaler



The Kerberos authentication occurs in the following stages:

Client authenticates itself to the KDC.

1. The NetScaler appliance receives a request from a client.
2. The traffic management (load balancing or content switching) virtual server on the NetScaler sends a challenge to the client.
3. To respond to the challenge, the client gets a Kerberos ticket.
   - The client sends the Authentication Server of the KDC a request for a ticket-granting ticket (TGT) and receives the TGT. (See 3, 4 in the figure, Kerberos Authentication Process.)
   - The client sends the TGT to the Ticket Granting Server of the KDC and receives a Kerberos ticket. (See 5, 6 in the figure, Kerberos Authentication Process.)

Note: The above authentication process is not necessary if the client already has a Kerberos ticket whose lifetime has not expired. In addition, clients such as Web Services, .NET, or J2EE, which support SPNEGO, get a Kerberos ticket for the target server, create an SPNEGO token, and insert the token in the HTTP header when they send an HTTP request. They do not go through the client authentication process.

Client requests a service.

1. The client sends the Kerberos ticket containing the SPNEGO token and the HTTP request to the traffic management virtual server on the NetScaler. The SPNEGO token has the necessary GSSAPI data.
2. The NetScaler establishes a security context between the client and the NetScaler. If the NetScaler cannot accept the data provided in the Kerberos ticket, the client is asked to get a different ticket. This cycle repeats till the GSSAPI data is acceptable and the security context is established. The traffic management virtual server on the NetScaler acts as an HTTP proxy between the client and the physical server.

NetScaler completes the authentication.

1. After the security context is complete, the traffic management virtual server validates the SPNEGO token.
2. From the valid SPNEGO token, the virtual server extracts the user ID and GSS credentials, and passes them to the authentication daemon.
3. A successful authentication completes the Kerberos authentication.

Kerberos Authentication - Configuration on the NetScaler Appliance

Mar 27, 2012

To configure Kerberos authentication on the NetScaler appliance, perform the following tasks:

1. Enable the Authentication, Authorization, and Auditing (AAA) feature on the NetScaler appliance.
2. On the Active Directory, add a user for Kerberos authentication, map the HTTP service to this user, and generate a keytab file and import it to the NetScaler appliance. You can map more than one service if the Kerberos authentication is required for more than one service. The keytab file should contain entries for every service that is bound to the traffic management virtual server on the NetScaler. The keytab file is necessary for decrypting the secret received from the client during Kerberos authentication. The authentication details of all the services are stored in a single keytab file on the NetScaler.
3. Add a DNS server.
   Note: The NetScaler must obtain the IP address of the domain controller from the fully qualified domain name (FQDN). Therefore, Citrix recommends configuring the NetScaler with a DNS server. A less preferred alternative is to create a static DNS entry.
4. Create an authentication negotiation policy with a negotiation action.
5. Configure an authentication server and bind the authentication policy to the authentication virtual server.
6. Configure an authentication service and a traffic management virtual server, and bind the service to the virtual server. You can use either a load balancing or a content switching virtual server.
7. Verify the configuration.

Enabling AAA on the NetScaler

Oct 30, 2013

Enable authentication of the traffic on the NetScaler appliance.

At the command prompt, type the following commands to enable AAA and verify the configuration:

- enable ns feature AAA
- show ns feature

Example

```
> enable feature aaa
 Done
> show ns feature
 Feature Acronym Status
 ------- ------- ------
1) Web Logging WL ON
 ...
3) Load Balancing LB ON
4) Content Switching CS ON
5) Cache Redirection CR ON
 ...
14) SSL VPN SSLVPN ON
15) AAA AAA ON
 ...
26) CloudBridge CloudBridge OFF
 Done
```

1. In the navigation pane, expand System, and then click Settings.
2. In the details pane, under Modes and Features, click Configure basic features.
3. In the Configure Basic Features dialog box, select the Authentication, Authorization and Auditing check box.
4. Click OK.
5. In the confirmation dialog box, click Yes. A message appears in the status bar to indicate that the feature is enabled.

Adding a Keytab file

Nov 14, 2013

The keytab file contains information about services necessary for Kerberos authentication. The keytab file is necessary for decrypting the secret received from the client during Kerberos authentication. You can map more than one service if the Kerberos authentication is required for more than one service. The keytab file should contain entries for every service that is bound to the traffic management virtual server on the NetScaler. The authentication details of all the services are stored in a single keytab file on the NetScaler.

To generate a keytab file and import it to the NetScaler appliance, follow the procedure described below:

Note: You can generate the keytab file and import it onto the NetScaler only from the command line.

1. Log onto the Active Directory server and create a user for Kerberos authentication.
   For example, type the following command:

   ```
   net user Kerb-SVC-Account freebsd!@#456 /add
   ```

2. In the User Properties section, ensure the following settings:
   - The Change password at next logon option is not selected.
   - The Password does not expire option is selected.
3. Map the HTTP service to the above user and export the keytab file. For example, run the following command on the Active Directory server:
   ```
   ktpass /out keytabfile /princ HTTP/owa.newacp.com@NEWACP.COM /pass freebsd!@#456 /mapuser newacp\dummy /ptype KRB5_NT_PRINCIPAL
   ```

   Note: If you want to map more services, repeat the above command for every service. You can give the same name or different names for the output file.
4. Transfer the keytab file to the NetScaler by using the unix ftp command or any other file transfer utility of your choice.
5. Log onto the NetScaler appliance, and run the ktutil utility to verify the keytab file. The keytab file has an entry for the HTTP service after it is imported.

```
root@ns# ktutil
ktutil: rkt /var/keytabfile
ktutil: list
slot KVNO Principal
----------------------------------------------------------------------

ktutil: wkt /etc/ krb5.keytab
ktutil: list
slot KVNO Principal
---- ---- ---------------------------------------------------------------
1 2 HTTP/owa.newacp.com@NEWACP.COM
ktutil: quit
```

Adding a DNS Server

Oct 30, 2013

The NetScaler appliance should obtain the IP address of the domain controller from the fully qualified domain name (FQDN). Therefore, Citrix recommends configuring the NetScaler with a DNS server. A less preferred alternative is to create a static DNS entry.

At the command prompt, type the following command:

add dns nameserver <IP>

Note: Alternatively, you can add static host entries or use any other means so that the NetScaler can resolve the FQDN name of the domain controller to an IP address.

Example

```
add dns nameserver 1.2.3.4
```

1. Navigate to Traffic Management > DNS > Name Servers.
2. In the details pane, click Add.
3. In the IP Address box, type the IP address.
4. Click Create, and then Close.
5. Verify that the details pane shows the newly added DNS server.

Creating an Authentication Negotiation Policy

Nov 14, 2013

Create a negotiation policy with a negotiation action for Kerberos authentication of services.

At the command prompt, type the following commands:

- add authentication negotiateAction <name> -domain <domainName> -domainUser <domainUsername> -domainUserPasswd <domainUserPassword> -encrypted
- add authentication negotiatePolicy <name> <rule> <reqAction>

Example

```
add authentication negotiateAction negact -domain newacp.com -domainUser Administrator -domainUserPasswd skp5sep
add authentication negotiatePolicy negopol ns_true negact
```

1. Navigate to Security > AAA-Application Traffic > Policies > Authentication.
2. In the details pane, on the Policies tab, click Add.
3. In the Create Authentication Policy dialog box, set the following parameters:
   - Name
   - Authentication Type - Select NEGOTIATE.
   - Server - Select an existing server from the dropdown list. To add a new authentication server, click New..., and in the Create Authentication Server dialog box, set the following parameters:
     - Domain Name
     - User Name
     - Password
     - Confirm Password - Retype the password.
   - Expression - In the Named Expression list, select General and select True Value from the dropdown list, and then click Add Expression.
4. Click Create, and then click Close.
5. Verify that the policy you created appears in the Authentication Policies and Servers pane.

Creating an Authentication Virtual Server

Oct 30, 2013

Configure an authentication virtual server and bind the authentication negotiation policy to the authentication virtual server.

At the command prompt, type the following commands:

- add authentication vserver <name> SSL <ipAuthVserver> 443 -authenticationDomain <domainName>
- bind authentication vserver <name> -policy <negotiatePolicyName>

Example

```
add authentication vserver authen1 SSL 10.102.113.166 443 -authenticationDomain newacp.com
add ssl certKey cert1 -cert "/nsconfig/ssl/complete/server/server_rsa_2048.pem" -key "/nsconfig/ssl/complete/server/server_rsa_2048.ky"
bind ssl vserver authen1 -certkeyName cert1
bind authentication vserver authen1 -policy negopol
```

1. Navigate to Security > AAA-Application Traffic > Virtual Servers.
2. In the details pane, click Add.
3. In the Create Virtual Server (Authentication) dialog box, set the following parameters:
   - Name
   - IP Address
   - Protocol - Select SSL
   - Domain - Type the fully qualified domain name added while creating the keytab file.

   Note: For AAA, the protocol must be SSL protocol and port must be 443. Therefore, these options are not provided.
4. On the Authentication tab, click Insert Policy. In the Authentication Policies group, from the Policy Name dropdown list, select the negotiate authentication policy you added for Kerberos authentication.
5. On the Certificates tab, select an SSL certificate from the list of available certificates, and then click Add. If the certificate you want to bind is not displayed in the Available Certificates list, click Install..., and then select the certificate file.
6. Click Create, and then click Close. The new authentication virtual server appears in the Authentication Virtual Servers pane.

Configuring a Traffic Management Virtual Server

Oct 30, 2013

Configure an authentication service and a traffic management virtual server, and bind the service to the virtual server. You can use either a load balancing or a content switching virtual server.

At the command prompt, type the following commands:

- add service <name>@ <ipBackendWebserver> HTTP 80
- add lb vserver <name>@ SSL <ipAddressLbVserver> 443 -authn401 ON -authnVsName <authVserverName>
- bind lb vserver <name>@ <serviceName>
  Note: Use a similar procedure for using a content switching virtual server as the traffic management virtual server.

Example

```
add service svc1 10.217.28.92 HTTP 80
 add lb vserver v2 HTTP 10.102.113.164 80 -persistenceType NONE -cltTimeout 180 -authn401 ON -authnVsName authen1
bind lb vserver v2 svc1
```

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, click Add.
3. In the Create Service dialog box, set the following parameters:
   - Service Name
   - Server
   - Protocol - Select HTTP.
   - Port - Select 80.
4. In the navigation pane, expand Load Balancing and click Virtual Servers.
5. In the details pane, click Add.
6. In the Create Virtual Server (Load balancing) dialog box, set values for the following parameters:
   - Name
   - IP Address
   - Protocol
   - Port
7. In the Create Virtual Server (Load balancing) dialog box, on the Services tab, select the service you created in Step 3 to Step 5.
8. In the Create Virtual Server (Load balancing) dialog box, on the Advanced tab, expand Authentication Settings, and then select the 401 Based Authentication check box.
9. Click Create, and then click Close. The new load balancing virtual server appears in the Load Balancing Virtual Servers pane.
10. In the details pane, verify the settings of the virtual server.
    Note: Use a similar procedure to create a content switching virtual server.
    Note: For more information, see Setting up basic load balancing.

Verifying the configuration for Kerberos Authentication

Mar 27, 2012

Ensure that you completed the following tasks and verify whether the configuration is complete and correct.

- Enable the AAA feature
- Import the keytab file
- Configure the DNS server
- Configure negotiation policies and actions
- Configure authentication virtual server
- Configure traffic management virtual server

To verify the configuration:

1. Access the load balancing virtual server, using the FQDN. For example, http://owa.newacp.com.
2. View the AAA session on the NetScaler. show aaa session

```
ClientIp (ClientPort) ->ServerIp(ServerPort)
 ---------------------------------------------
 PE id : 4
 User name: john.smith@NEWACP.COM Session Type: TM
Done
```

Configuration of Kerberos Authentication on a Client

Mar 27, 2012

Kerberos support must be configured on the browser to use Kerberos for authentication. You can use any Kerberos-compliant browser. Instructions for configuring Kerberos support on Internet Explorer and Mozilla Firefox follow. For other browsers, see the documentation of the browser.

1. In the Tools menu select Internet Options.
2. On the Security tab, click Local Intranet, and then click Sites.
3. In the Local Intranet dialog box, make sure that the Automatically detect intranet network option is selected, and then click Advanced.
4. In the Local Intranet dialog box, add the web sites of the domains of the traffic management virtual server on the NetScaler. The specified sites become local intranet sites.
5. Click Close or OK to close the dialog boxes.

1. Make sure that you have Kerberos properly configured on your computer.
2. Type about:config in the URL bar.
3. In the filter text box, type network.negotiate.
4. Change network.negotiate-auth.delegation-uris to the domain that you want to add.
5. Change network.negotiate-auth.trusted-uris to the domain that you want to add.
   Note: If you are running Windows, you also need to enter sspi in the filter text box and change the network.auth.use-sspi option to False.

Offloading Kerberos Authentication from Physical Servers

Nov 14, 2013

The NetScaler appliance can offload authentication tasks from servers. Instead of the physical servers authenticating the requests from clients, the Netscaler authenticates all the client requests before it forwards them to any of the physical servers bound to it. The user authentication is based on Active Directory tokens.

There is no authentication between the NetScaler and the physical server, and the authentication offload is transparent to the end users. After the initial logon to a Windows computer, the end user does not have to enter any additional authentication information in a pop-up or on a logon page.

In the current NetScaler release, Kerberos authentication is available only for Authentication, Authorization, and Auditing (AAA) Traffic Management Virtual Servers. Kerberos authentication is not supported for SSL VPN in the NetScaler Gateway Enterprise Edition appliance or for NetScaler appliance management.

Kerberos authentication requires configuration on the NetScaler appliance and on client browsers.

1. Create a user account on Active Directory. When creating a user account, verify the following options in the User Properties section:
   - Make sure that you do not select the Change password at next logon option.
   - Be sure to select the Password does not expire option.
2. On the NetScaler appliance, at the CLI command prompt, type:
   - ktpass -princ HTTP/kerberos.crete.lab.net@crete.lab.net -ptype KRB5_NT_PRINCIPAL -mapuser kerbuser@crete.lab.net -mapop set -pass Citrix1 -out C:\kerbtabfile.txt

   Note: Be sure to type the above command on a single line. The output of the above command is written into the C:\kerbatabfile.txt file.
3. Upload the kerbatabfile.txt file to the /etc directory of the NetScaler appliance by using a Secure Copy (SCP) client.
4. Run the following command to add a DNS server to the NetScaler appliance.
   - add dns nameserver 1.2.3.4

   The NetScaler appliance cannot process Kerberos requests without the DNS server. Be sure to use the same DNS server that is used in the Microsoft Windows domain.
5. Switch to the shell prompt and run the following commands from the shell prompt:
   - ktutil # rkt /etc/kerbtabfile.txt
   - # wkt /etc/krb5.keytab
   - # list

     The list command displays the user account details that you created in the Active Directory. A sample screen of the output of the list command is shown below.

     Figure 1. Sample Output of the list Command

```
> shell
Copyright (c) 1992-2008 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992,
        The Regents of the University of California. All righ

root@ns# cd /etc
root@ns# ls -la *.txt
-rw-r--r--  1 root  wheel  82 Apr  4 00:43 kerbtabfile.txt
root@ns# ktutil
ktutil:  rkt /etc/kerbtabfile.txt
ktutil:  wkt /etc/krb5.keytab
ktutil:  list
slot KVNO Principal
---- ---- ---------------------------------------------------
   1    3 HTTP/kerberos.crete.example.com@crete.example.com
ktutil:  quit
root@ns#
```

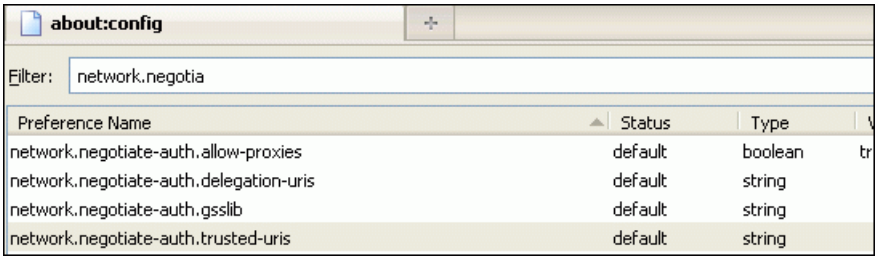
6. Switch to the command line interface of NetScaler.
7. Run the following command to create a Kerberos authentication server:
   - add authentication negotiateAction KerberosServer -domain "crete.lab.net" -domainUser kerbuser -domainUserPasswd Citrix1
8. Run the following command to create a negotiation policy:
   - add authentication negotiatePolicy Kerberos-Policy "REQ.IP.DESTIP == 192.168.17.200" KerberosServer
9. Run the following command to create an authentication virtual server.
   - add authentication vserver Kerb-Auth SSL 192.168.17.201 443 -AuthenticationDomain crete.example.com
10. Run the following command to bind the Kerberos policy to the authentication virtual server:
    - bind authentication vserver Kerb-Auth -policy Kerberos-Policy -priority 100
11. Run the following command to bind an SSL certificate to the authentication virtual server. You can use one of the test certificates, which you can install from the GUI NetScaler appliance. Run the following command to use the ServerTestCert sample certificate.
    - bind ssl vserver Kerb-Auth -certkeyName ServerTestCert
12. Create an HTTP load balancing virtual server with the IP address, 192.168.17.200.
    Ensure that you create a virtual server from the command line interface for NetScaler 9.3 releases if they are older than 9.3.47.8.
13. Run the following command to configure an authentication virtual server:
    - set lb vserver <name>-authn401 ON -authnVsName Kerb-Auth
14. Enter the host name http://www.crete.example.com in the address bar of the Web browser.
    The Web browser displays an authentication dialog box because the Kerberos authentication is not set up in the browser.

    Note: Kerberos authentication requires a specific configuration on the client. Ensure that the client can resolve the hostname, which results in the Web browser connecting to an HTTP virtual server.
15. Configure Kerberos on the Web browser of the client computer.
    - For configuring on Internet Explorer, see "Configuring Internet Explorer for Kerberos authentication."
    - For configuring on Mozilla Firefox, see "Configuring Mozilla Firefox for Kerberos authentication."
16. Verify whether you can access the backend physical server without authentication.

1. Select Internet Options from the Tools menu.
2. Activate the Security tab.
3. Select Local Intranet from the Select a zone to view change security settings section.
4. Click Sites.

5. Click Advanced.
6. Specify the URL, http://www.crete.example.com and click Add.
7. Restart Internet Explorer.

1. Enter about:config in the address bar of the browser.
2. Click the warning disclaimer.
3. Type Network.Negotiate-auth.trusted-uris in the Filter box.
4. Double click Network.Negotiate-auth.trusted-uris. A sample screen is shown below.



5. In the Enter String Value dialog box, specify www.crete.example.com.
6. Restart Firefox.

NetScaler Kerberos Single Sign-On

Oct 11, 2013

NetScaler appliances now support single sign-on (SSO) using the Kerberos 5 protocol. Users log on to a proxy, the Application Delivery Controller (ADC), which then provides access to protected resources.

The NetScaler Kerberos SSO implementation requires the user's password for SSO methods that rely on basic, NTLM, or forms-based authentication. The user's password is not required for Kerberos SSO, although if Kerberos SSO fails and the NetScaler appliance has the user's password, it uses the password to attempt NTLM SSO.

If the user's password is available, the KCD account is configured with a realm, and no delegated user information is present, the NetScaler Kerberos SSO engine impersonates the user to obtain access to authorized resources. Impersonation is also called unconstrained delegation.

The NetScaler Kerberos SSO engine can also be configured to use a delegated account to obtain access to protected resources on the user's behalf. This configuration requires delegated user credentials, a keytab, or a delegated user certificate and matching CA certificate. Configuration that uses a delegated account is called constrained delegation.

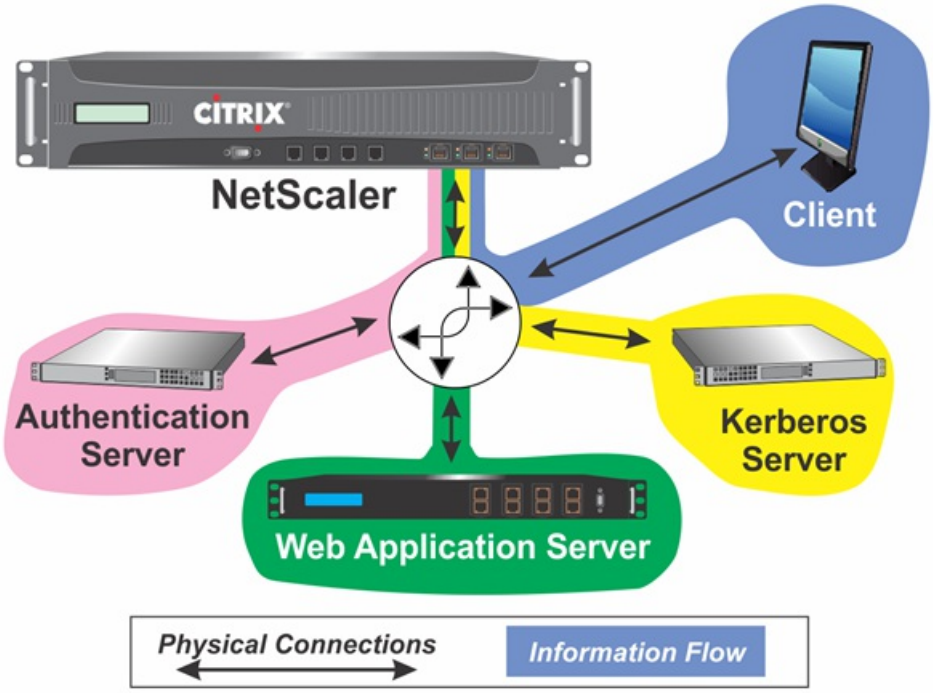
An Overview of NetScaler Kerberos SSO

Oct 14, 2013

To use the NetScaler Kerberos SSO feature does, users first authenticate with Kerberos or a supported third-party authentication server. Once authenticated, the user requests access to a protected web application. The web server responds with a request for proof that the user is authorized to access that web application. The user's browser contacts the Kerberos server, which verifies that the user is authorized to access that resource, and then provides the user's browser with a service ticket that provides proof. The browser resends the user's request to the web application server with the service ticket attached. The web application server verifies the service ticket, and then allows the user to access the application.

AAA-TM implements this process as shown in the following diagram. The diagram illustrates the flow of information through the NetScaler appliance and AAA-TM, on a secure network with LDAP authentication and Kerberos authorization. AAA-TM environments that use other types of authentication have essentially the same information flow, although they might differ in some details.

Figure 1. A Secure Network with LDAP and Kerberos



NetScaler AAA-TM authentication and authorization in a Kerberos environment requires that the following actions take place.

1. The client sends a request for a resource to the traffic management virtual server on the NetScaler appliance.
2. The traffic management virtual server passes the request to the authentication virtual server, which authenticates the client and then passes the request back to the traffic management virtual server.
3. The traffic management virtual server sends the client's request to the web application server.
4. The web application server responds to the traffic management virtual server with a 401 Unauthorized message that requests Kerberos authentication, with fallback to NTLM authentication if the client does not support Kerberos.
5. The traffic management virtual server contacts the Kerberos SSO daemon.

6. The Kerberos SSO daemon contacts the Kerberos server and obtains a ticket-granting ticket (TGT) allowing it to request service tickets authorizing access to protected applications.
7. The Kerberos SSO daemon obtains a service ticket for the user and sends that ticket to the traffic management virtual server.
8. The traffic management virtual server attaches the ticket to the user's initial request and sends the modified request back to the web application server.
9. The web application server responds with a 200 OK message.

These steps are transparent to the client, which just sends a request and receives the requested resource.

All AAA-TM authentication mechanisms support NetScaler Kerberos SSO. AAA-TM supports the Kerberos SSO mechanism with the Kerberos, CAC (Smart Card) and SAML authentication mechanisms with any form of client authentication to the NetScaler appliance. It also supports the HTTP-Basic, HTTP-Digest, Forms-based, and NTLM (versions 1 and 2) SSO mechanisms if the client uses either HTTP-Basic or Forms-Based authentication to log on to the NetScaler appliance.

The following table shows each supported client-side authentication method, and the supported server-side authentication method for that client-side method.

Table 1. Supported Authentication Methods

	Basic/Digest/NTLM	Kerberos Constrained Delegation	User Impersonation
CAC (Smart Card): at SSL/TLS Layer		X	X
Forms-Based (LDAP/RADIUS/TACACS)	X	X	X
HTTP Basic (LDAP/RADIUS/TACACS)	X	X	X
Kerberos		X	
NTLM v1/v2		X	X
SAML		X	
SAML Two-Factor	X	X	X
Certificate Two-Factor	X	X	X

Setting up NetScaler SSO

Oct 08, 2013

You can configure NetScaler SSO to work in one of two ways: by impersonation or by delegation. SSO by impersonation is a simpler configuration than SSO by delegation, and is therefore usually preferable when your configuration allows it. To configure NetScaler SSO by impersonation, you must have the user's user name and password.

To configure NetScaler SSO by delegation, you must have the delegated user's credentials in one of the following formats: the user's user name and password, the keytab configuration that includes the user name and an encrypted password, or the delegated user certificate and the matching CA certificate.

Prerequisites

Mar 30, 2015

Before you configure NetScaler SSO, you need to have your NetScaler appliance fully configured to manage traffic to and authentication for your web application servers. Therefore, you must configure either load balancing or content switching, and then AAA, for these web application servers. You should also verify routing between the appliance, your LDAP server, and your Kerberos server.

If your network is not already configured in this manner, perform the following configuration tasks:

- Configure a server and service for each web application server.
- Configure a traffic management virtual server to handle traffic to and from your web application server.

Following are brief instructions and examples for performing each of these tasks from the NetScaler command line. For further assistance, see "Setting Up Basic Load Balancing" and "Setting up AAA Virtual Servers and DNS."

For NetScaler SSO to obtain a TGS (service ticket) for a service, either the FQDN assigned to the server entity on the NetScaler appliance must match the FQDN of the web application server, or the server entity name must match the NetBios name of the web application server. You can take either of the following approaches:

- Configure the NetScaler server entity by specifying the FQDN of the web application server.
- Configure the NetScaler server entity by specifying the IP address of the web application server, and assign the server entity the same name as the NetBios name of the web application server.

At the command prompt, type the following commands:

- add server name <serverFQDN>
- add service name serverName serviceType port

For the variables, substitute the following values:

- **serverName**—A name for the NetScaler appliance to use to refer to this server.
- **serverFQDN**—The FQDN of the server. If the server has no domain assigned to it, use the server's IP address and make sure that the server entity name matches the NetBios name of the web application server.
- **serviceName**—A name for the NetScaler appliance to use to refer to this service.
- **type**—The protocol used by the service, either HTTP or MSSQLSVC.
- **port**—The port on which the service listens. HTTP services normally listen on port 80. Secure HTTPS services normally listen on port 443.

The following examples add server and service entries on the NetScaler appliance for the web application server **was1.example.com**. The first example uses the FQDN of the web application server; the second uses the IP address.

To add the server and service using the web application server FQDN, **was1.example.com**, you would type the following commands:

```
add server was1 was1.example.com
add service was1service was1 HTTP 80
```

To add the server and service using the web application server IP and NetBios name, where the web application server IP is 10.237.64.87 and its NetBios name is WAS1, you would type the following commands:

```
add server WAS1 10.237.64.87
add service was1service WAS1 HTTP 8
```

The traffic management virtual server manages traffic between the client and the web application server. You can use either a load balancing or a content switching virtual server as the traffic management server. The SSO configuration is the same for either type.

To create a load balancing virtual server, at the command prompt, type the following command:

add lb vserver <vserverName> <type> <IP> <port>
For the variables, substitute the following values:

- **vserverName**—A name for the NetScaler appliance to use to refer to this virtual server.
- **type**—The protocol used by the service, either HTTP or MSSQLSVC.
- **IP**—The IP address assigned to the virtual server. This would normally be an IANA-reserved, non-public IP address on your LAN.
- **port**—The port on which the service listens. HTTP services normally listen on port 80. Secure HTTPS services normally listen on port 443.

To add a load balancing virtual server called tmvserver1 to a configuration that manages HTTP traffic on port 80, assigning it a LAN IP address of 10.217.28.20 and then binding the load balancing virtual server to the wasservice1 service, you would type the following commands:

```
add lb vserver tmvserver1 HTTP 10.217.28.20 80
bind lb vserver tmvserv1 wasservice1
```

The authentication virtual server manages authentication traffic between the client and the authentication (LDAP) server. To create an authentication virtual server, at the command prompt type the following commands:

- add authentication vserver <authvserverName> SSL <IP> 443
- set authentication vserver <authvservername> –authenticationdomain <domain>

For the variables, substitute the following values:

- **authvserverName** —A name for the NetScaler appliance to use to refer to this authentication virtual server. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the authentication virtual server is added by using the rename authentication vserver command.
- **IP**—The IP address assigned to the authentication virtual server. As with the traffic management virtual server, this address would normally be an IANA-reserved, non-public IP on your LAN.
- **domain**—The domain assigned to the virtual server. This would usually be the domain of your network. It is customary, though not required, to enter the domain in all capitals when configuring the authentication virtual server.

To add an authentication virtual server called authverver1 to your configuration and assign it the LAN IP 10.217.28.21 and the domain EXAMPLE.COM, you would type the following commands:

```
add authentication vserver authvserver1 SSL 10.217.28.21 443
set authentication vserver authvserver1 –authenticationdomain EXAMPLE.COM
```

The authentication virtual server can be configured to handle authentication for a single domain or for multiple domains. If it is configured to support authentication for multiple domains, you must also specify the domain for NetScaler SSO by creating an authentication profile, and then configuring the traffic management virtual server to use that authentication profile.

Note: The traffic management virtual server can be either a load balancing (lb) or content switching (cs) virtual server. The following instructions assume that you are using a load balancing virtual server. To configure a content switching virtual server, simply substitute set cs vserver for set lb vserver. The procedure is otherwise the same.
To create the authentication profile, and then configure the authentication profile on a traffic management virtual server, type the following commands:

- add authentication authnProfile <authnProfileName> {-authvserverName <string>} {-authenticationHost <string>} {-authenticationDomain <string>}
- set lb vserver <vserverName> -authnProfile <authnprofileName>

For the variables, substitute the following values:

- authnprofileName—A name for the authentication profile. Must begin with a letter, number, or the underscore character (_), and must consist of from one to thirty-one alphanumeric or hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters.
- authvserverName—The name of the authentication virtual server that this profile uses for authentication.
- authenticationHost—Host name of the authentication virtual server.
- authenticationDomain—Domain for which NetScaler SSO handles authentication. Required if the authentication virtual server performs authentication for more than one domain, so that the correct domain is included when the NetScaler appliance sets the traffic management virtual server cookie.

To create an authentication profile named authnProfile1 for authentication of the example.com domain, and to configure the load balancing virtual server vserver1 to use the authentication profile authnProfile1, you would type the following commands:

```
add authentication authnProfile authnProfile1 -authnvsName authvsesrver1
    -authenticationHost authvsesrver1 -authenticationDomain example.com
set lb vserver vserver1 -authnProfile authnProfile1
```

Configuring SSO

Oct 13, 2013

Configuring NetScaler SSO to authenticate by impersonation is simpler than configuring than SSO to authenticate by delegation, and is therefore usually preferable when your configuration allows it. You just create a KCD account. You can use the user's password.

If you do not have the user's password, you can configure NetScaler SSO to authenticate by delegation. Although somewhat more complex than configuring SSO to authenticate by impersonation, the delegation method provides flexibility in that a user's credentials might not be available to the NetScaler appliance in all circumstances.

For either impersonation or delegation, you must also enable integrated authentication on the web application server.

Enabling Integrated Authentication on the Web Application Server

Oct 14, 2013

To set up NetScaler Kerberos SSO on each web application server that Kerberos SSO will manage, use the configuration interface on that server to configure the server to require authentication. Select Kerberos (negotiate) authentication by preference, with fallback to NTLM for clients that do not support Kerberos.

Following are instructions for configuring Microsoft Internet Information Server (IIS) to require authentication. If your web application server uses software other than IIS, consult the documentation for that web server software for instructions.

1. Log on to the IIS server and open Internet Information Services Manager.
2. Select the web site for which you want to enable integrated authentication. To enable integrated authentication for all IIS web servers managed by IISM, configure authentication settings for the Default Web Site. To enable integrated authentication for individual services (such as Exchange, Exadmin, ExchWeb, and Public), configure these authentication settings for each service individually.
3. Open the Properties dialog box for the default web site or for the individual service, and click the Directory Security tab.
4. Beside Authentication and Access Control, select Edit.
5. Disable anonymous access.
6. Enable Integrated Windows authentication (only). Enabling integrated Windows authentication should automatically set protocol negotiation for the web server to **Negotiate, NTLM**, which specifies Kerberos authentication with fallback to NTLM for non-Kerberos capable devices. If this option is not automatically selected, manually set protocol negotiation to **Negotiate, NTLM**.

Setting Up SSO by Impersonation

Mar 18, 2015

You can configure the KCD account for NetScaler SSO by impersonation. In this configuration, the NetScaler appliance obtains the user's username and passwordwhen the user authenticates to the authentication server and uses those credentials to impersonate the user to obtain a ticket-granting ticket (TGT). If the user's name is in UPN format, the appliance obtains the user's realm from UPN. Otherwise, it obtains the user's name and realm by extracting it from the SSO domain used during initial authentication, or from the session profile.

When configuring the KCD account, you must set the realm parameter to the realm of the service that the user is accessing. The same realm is also used as the user's realm if the user's realm cannot be obtained from authentication with the Netscaler appliance or from the session profile.

At the command prompt, type the following command:

add aaa kcdaccount <accountname> -realmStr <realm>
For the variables, substitute the following values:

- **accountname**—The KCD account name.
- **realm**—The domain assigned to NetScaler SSO.

Example:

To add a KCD account named **kcdccount1**, and use the keytab named **kcdvserver.keytab**, you would type the following command:

add aaa kcdAccount kcdaccount1 -keytab kcdvserver.keytab

Configuring SSO by Delegation

Oct 13, 2013

To configure SSO by Delegation, you need to perform the following tasks:

- If you are configuring delegation by delegated user certificate, install the matching CA certificates on the NetScaler appliance and add them to the NetScaler configuration.
- Create the KCD account on the appliance. The appliance uses this account to obtain service tickets for your protected applications.
- Configure the Active Directory server.

If you are configuring NetScaler SSO with a client certificate, you must copy the matching CA certificate for the client certificate domain (the client CA certificate) to the NetScaler appliance, and then install the CA certificate. To copy the client CA certificate, use the file transfer program of your choice to transfer the certificate and private-key file to the NetScaler appliance, and store the files in **/nsconfig/ssl**.

To install the client CA certificate on the NetScaler appliance

At the command prompt, type the following command:

add ssl certKey <certkeyName> -cert <cert> [(-key <key> [-password]) | -fipsKey <fipsKey>] [-inform (DER | PEM)] [-expiryMonitor (ENABLED | DISABLED | UNSET) [-notificationPeriod <positive_integer>]] [-bundle (YES | NO)]
For the variables, substitute the following values:

- **certkeyName**—A name for the client CA certificate. Must begin with an ASCII alphanumeric or underscore (_) character, and must consist of from one to thirty-one characters. Allowed characters include the ASCII alphanumerics, underscore, hash (#), period(.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the certificate-key pair is created. If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cert" or 'my cert').
- **cert**—Full path name and file name of the X509 certificate file used to form the certificate-key pair. The certificate file must be stored on the NetScaler appliance, in the /nsconfig/ssl/ directory.
- **key**—Full path name and file name of the file that contains the private key to the X509 certificate file. The key file must be stored on the NetScaler appliance in the /nsconfig/ssl/ directory.
- **password**—If a private key is specified, the passphrase used to encrypt the private key. Use this option to load encrypted private keys in PEM format.
- **fipsKey**—Name of the FIPS key that was created inside the Hardware Security Module (HSM) of a FIPS appliance, or a key that was imported into the HSM.
  Note: You can specify either a **key** or a **fipsKey**, but not both.
- **inform**—Format of the certificate and private-key files, either PEM or DER.
- **passplain**—Pass phrase used to encrypt the private key. Required when adding an encrypted private-key in PEM format.
- **expiryMonitor**—Configure the NetScaler appliance to issue an alert when the certificate is about to expire. Possible values: ENABLED, DISABLED, UNSET.
- **notificationPeriod**—If expiryMonitor is ENABLED, number of days before the certificate expires to issue an alert.
- **bundle**—Parse the certificate chain as a single file after linking the server certificate to its issuer's certificate within the file. Possible values: YES, NO.

The following example adds the specified delegated user certificate **customer-cert.pem** to the NetScaler configuration along with the key **customer-key.pem**, and sets the password, certificate format, expiration monitor, and notification period.

To add the delegated user certificate, you would type the following commands:

```
add ssl certKey customer -cert "/nsconfig/ssl/customer-cert.pem"
    -key "/nsconfig/ssl/customer-key.pem" -password "dontUseDefaultPWs!"
    -inform PEM -expiryMonitor ENABLED [-notificationPeriod 14
```

If you are configuring NetScaler SSO by delegation, you can configure the KCD account to use the user's log-on name and password, to use the user's log-on name and keytab, or to use the user's client certificate. If you configure SSO with user name and password, the NetScaler appliance uses the delegated user account to obtain a Ticket Granting Ticket (TGT), and then uses the TGT to obtain service tickets for the specific services that each user requests. If you configure SSO with keytab file, the NetScaler appliance uses the delegated user account and keytab information. If you configure SSO with a delegated user certificate, the NetScaler appliance uses the delegated user certificate.

To create the KCD account for SSO by delegation with a password

At the command prompt, type the following commands:

add aaa kcdaccount <accountname> -delegatedUser root -kcdPassword <password> -realmStr <realm>
For the variables, substitute the following values:

- **accountname**—A name for the KCD account.
- **password**—A password for the KCD account.
- **realm**—The realm of the KCD account, usually the domain for which SSO is active.

To add a KCD account named **kcdaccount1** to the NetScaler appliance configuration with a password of **password1** and a realm of **EXAMPLE.COM**, specifying the delegated user account in UPN format (as root), you would type the following commands:

```
add aaa kcdaccount kcdaccount1 –delegatedUser root
-kcdPassword password1 -realmStr EXAMPLE.COM
```

To add a KCD account named **kcdaccount1** to the NetScaler appliance configuration with a password of **password1** and a realm of **EXAMPLE.COM**, specifying the delegated user account in SPN format, you would type the following commands:

```
add aaa kcdAccount kcdaccount1 -realmStr EXAMPLE.COM
-delegatedUser "host/kcdvserver.example.com" -kcdPassword password1
```

Creating the KCD account for SSO by delegation with a keytab

If you plan to use a keytab file for authentication, first create the keytab. You can create the keytab file manually by logging onto the AD server and using the ktpass utility, or you can use the NetScaler configuration utility to create a batch script, and then run that script on the AD server to generate the keytab file. Next, use FTP or another file transfer program

to transfer the keytab file to the NetScaler appliance and place it in the /nsconfig/krb directory. Finally, configure the KCD account for NetScaler SSO by delegation and provide the path and file name of the keytab file to the NetScaler appliance.

Log on to the AD server command line and, at the command prompt, type the following command:

ktpass /princ <SPN> /ptype KRB5_NT_PRINCIPAL /mapuser <DOMAIN>\<username> /pass <password> -out <File_Path>
For the variables, substitute the following values:

- **SPN**—The service principal name for the KCD service account.
- **DOMAIN**—The domain of the Active Directory server.
- **username**—The KSA account username.
- **password**—The KSA account password.
- **path**— The full path name of the directory in which to store the keytab file after it is generated.

1. Navigate to Security > AAA - Application Traffic
2. In the data pane, under Kerberos Constrained Delegation, click Batch file to generate Keytab.
3. In the Generate KCD (Kerberos Constrained Delegation) Keytab Script dialog box, set the following parameters:
   - **Domain User Name**—The KSA account username.
   - **Domain Password**—The KSA account password.
   - **Service Principal**—The service principal name for the KSA.
   - **Output File Name**—The full path and file name to which to save the keytab file on the AD server.
4. Clear the Create Domain User Account check box.
5. Click Generate Script.
6. Log on to the Active Directory server and open a command line window.
7. Copy the script from the Generated Script window and paste it directly into the Active Directory server command-line window. The keytab is generated and stored in the directory under the file name that you specified as **Output File Name**.
8. Use the file transfer utility of your choice to copy the keytab file from the Active Directory server to the NetScaler appliance and place it in the **/nsconfig/krb** directory.

At the command prompt, type the following command:

add aaa kcdaccount <accountname> –keytab <keytab>
Example:

To add a KCD account named **kcdccount1**, and use the keytab named **kcdvserver.keytab**, you would type the following commands:

add aaa kcdaccount kcdaccount1 –keytab kcdvserver.keytab

To create the KCD account for SSO by delegation with a delegated user cert

At the command prompt, type the following command:

add aaa kcdaccount <accountname> -realmStr <realm> -delegatedUser <user_name/SPN> -usercert <cert> -cacert <cacert>
For the variables, substitute the following values:

- **accountname**—A name for the KCD account.
- **realmStr**—The realm for the KCD account, usually the domain for which SSO is configured.
- **delegatedUser**—The delegated user name, in SPN format.
- **usercert**—The full path and name of the delegated user certificate file on the NetScaler appliance. The delegated user certificate must contain both the client certificate and the private key, and must be in PEM format. If you use smart card authentication, you might need to create a smart card certificate template to allow certificates to be imported with the private key.
- **cacert**—The full path to and name of the CA certificate file on the NetScaler appliance.

To add a KCD account named **kcdccount1**, and use the keytab named **kcdvserver.keytab**, you would type the following command:

```
add aaa kcdaccount kcdaccount1 -realmStr EXAMPLE.COM
    -delegatedUser "host/kcdvserver.example.com" -usercert /certs/usercert
    -cacert /cacerts/cacert
```

When you configure SSO by delegation, in addition to creating the KCDAccount on the NetScaler appliance, you must also create a matching Kerberos Service Account (KSA) on your LDAP active directory server, and configure the server for SSO. To create the KSA, use the account creation process on the active directory server. To configure SSO on the active directory server, open the properties window for the KSA. In the Delegation tab, enable the following options: Trust this user for delegation to specified services only and Use any Authentication protocol. (The Kerberos only option does not work, because it does not enable protocol transition or constrained delegation.) Finally, add the services that NetScaler SSO will manage.

Note: If the Delegation tab is not visible in the KSA account properties dialog box, before you can configure the KSA as described, you must use the Microsoft **setspn** command-line tool to configure the active directory server so that the tab is visible.

To configure delegation for the Kerberos service account

1. In the LDAP account configuration dialog box for the Kerberos service account that you created, click the Delegation tab.
2. Choose "Trust this user for delegation to the specified services only".
3. Under "Trust this user for delegation to the specified services only," choose "Use any authentication protocol".
4. Under “Services to which this account can present delegated credentials,” click Add.
5. In the Add Services dialog box, click Users or Computers, choose the server that hosts the resources to be assigned to the service account, and then click OK.
   Note: Constrained delegation does not support services hosted in domains other than the domain assigned to the account, even though Kerberos might have a trust relationship with other domains
6. Back in the Add Services dialog box, in the Available Services list, choose the services assigned to the service account. NetScaler SSO supports the HTTP and MSSQLSVC services.
7. Click OK.

Application Firewall

Mar 28, 2012

The following topics cover installation and configuration of the Citrix Application Firewall feature.

Introduction	An overview of web application security and how the application firewall works.
Configuration	How to configure the application firewall to protect a web site, a web service, or a web 2.0 site.
Signatures	A detailed description of the signatures feature and how to configure the signatures, add signatures from a supported vulnerability scanning tool, and define your own signatures, with examples.
Advanced Protections	A detailed description of all of the application firewall security checks, with configuration information and examples.
Profiles	A description of how profiles are configured and used in the application firewall.
Policies	A description of how policies are used when configuring the application firewall, with examples of useful policies.
Imports	A description of how the application firewall uses different types of imported files, and how to import and export files.
Global Configuration	A description of application firewall features that apply to all profiles, and how to configure them.
Use Cases	Extended examples that demonstrate how to set up the application firewall to best protect specific types of more complex web sites and web services.
Logs, Statistics, and Reports	How to access and use the application firewall logs, the statistics, and the reports to assist in configuring the application firewall.

Introduction

Oct 08, 2014

The Citrix NetScaler Application Firewall prevents security breaches, data loss, and possible unauthorized modifications to web sites that access sensitive business or customer information. It does so by filtering both requests and responses, examining them for evidence of malicious activity, and blocking those that exhibit such activity. Your site is protected not only from common types of attacks, but also from new, as yet unknown attacks. In addition to protecting web servers and web sites from unauthorized access and misuse by hackers and malicious programs, the application firewall provides protection against security vulnerabilities in legacy CGI code or scripts, other web frameworks, web server software, and the underlying operating systems.

The NetScaler Application Firewall is available as a stand-alone appliance, or as a feature on a Citrix NetScaler application delivery controller (ADC) or Citrix NetScaler virtual appliance (VPX). In the application firewall documentation, the term NetScaler ADC refers to the platform on which the application firewall is running, regardless of whether that platform is a dedicated firewall appliance, a NetScaler ADC on which other features have also been configured, or a NetScaler VPX.

To use the application firewall, you must create at least one security configuration to block connections that violate the rules that you set for your protected web sites. The number of security configurations that you might want to create depends on the complexity of your web site. In some cases, a single configuration is sufficient. In other cases, particularly those that include interactive web sites, web sites that access database servers, online stores with shopping carts, you might need several different configurations to best protect sensitive data without wasting significant effort on content that is not vulnerable to certain types of attacks. You can often leave the defaults for the global settings, which affect all security configurations, unchanged. However, you can change the global settings if they conflict with other parts of your configuration or you prefer to customize them.

Web Application Security

Oct 08, 2014

Web application security is network security for computers and programs that communicate by using the HTTP and HTTPS protocols. This is an extremely broad area in which security flaws and weaknesses abound. Operating systems on both servers and clients have security issues and are vulnerable to attack. Web server software and web site enabling technologies such as CGI, Java, JavaScript, PERL and PHP have underlying vulnerabilities. Browsers and other client applications that communicate with web-enabled applications also have vulnerabilities. Web sites that use any technology but the simplest of HTML, including any site that allows interaction with visitors, often have vulnerabilities of their own.

In the past, a breach in security was often just an annoyance, but today that is seldom the case. For example, attacks in which a hacker gained access to a web server and made unauthorized modifications to (defaced) a web site used to be common. They were usually launched by hackers who had no motivation beyond demonstrating their skills to fellow hackers or embarrassing the targeted person or company. Most current security breaches, however, are motivated by a desire for money. The majority attempt to accomplish one or both of the following goals: to obtain sensitive and potentially valuable private information, or to obtain unauthorized access to and control of a web site or web server.

Certain forms of web attacks focus on obtaining private information. These attacks are often possible even against web sites that are secure enough to prevent an attacker from taking full control. The information that an attacker can obtain from a web site can include customer names, addresses, phone numbers, social security numbers, credit card numbers, medical records, and other private information. The attacker can then use this information or sell it to others. Much of the information obtained by such attacks is protected by law, and all of it by custom and expectation. A breach of this type can have extremely serious consequences for customers whose private information is compromised. At best, these customers will have to exercise vigilance to prevent others from abusing their credit cards, opening unauthorized credit accounts in their name, or appropriating their identities outright (identity theft). At worst, the customers may face ruined credit ratings or even be blamed for criminal activities in which they had no part.

Other web attacks are aimed at obtaining control of (or *compromising*) a web site or the server on which it operates, or both. A hacker who gains control of a web site or server can use it to host unauthorized content, act as a proxy for content hosted on another web server, provide SMTP services to send unsolicited bulk email, or provide DNS services to support such activities on other compromised web servers. Most web sites that are hosted on compromised web servers promote questionable or outright fraudulent businesses. For example, the majority of phishing web sites and child exploitation web sites are hosted on compromised web servers.

Protecting your web sites and web services against these attacks requires a multilayered defense capable of both blocking known attacks with identifiable characteristics and protecting against unknown attacks, which can often be detected because they look different from the normal traffic to your web sites and web services.

Updated: 2014-10-08

The first line of defense for your web sites is protection against the large number of attacks that are known to exist and have been observed and analyzed by web security experts. Common types of attacks against HTML-based web sites include:

- **Buffer overflow attacks.** Sending an extremely long URL, extremely long cookie, or other extremely long bit of information to a web server in hopes of causing it or the underlying operating system to hang, crash, or provide the

attacker with access to the underlying operating system. A buffer overflow attack can be used to gain access to unauthorized information, to compromise a web server, or both.

- **Cookie security attacks.** Sending a modified cookie to a web server, usually in hopes of obtaining access to unauthorized content by using falsified credentials.
- **Forceful browsing.** Accessing URLs on a web site directly, without navigating to the URLs by means of hyperlinks on the home page or other common start URLs on the web site. Individual instances of forceful browsing may simply indicate a user who bookmarked a page on your web site, but repeated attempts to access nonexistent content, or content that users should never access directly, often represent an attack on web site security. Forceful browsing is normally used to gain access to unauthorized information, but can also be combined with a buffer overflow attack in an attempt to compromise your server.
- **Web form security attacks.** Sending inappropriate content to your web site in a web form. Inappropriate content can include modified hidden fields, HTML or code in a field intended for alphanumeric data only, an overly long string in a field that accepts only a short string, an alphanumeric string in a field that accepts only an integer, and a wide variety of other data that your web site does not expect to receive in that web form. A web form security attack can be used either to obtain unauthorized information from your web site or to compromise the web site outright, usually when combined with a buffer overflow attack.

Two specialized types of attacks on web form security deserve special mention:

- **SQL injection attacks.** Sending an active SQL command or commands in a web form or as part of a URL, with the goal of causing an SQL database to execute the command or commands. SQL injection attacks are normally used to obtain unauthorized information.
- **Cross-site scripting attacks.** Using a URL or a script on a web page to violate the same-origin policy, which forbids any script from obtaining properties from or modifying any content on a different web site. Since scripts can obtain information and modify files on your web site, allowing a script access to content on a different web site can provide an attacker the means to obtain unauthorized information, to compromise a web server, or both.

Attacks against XML-based web services normally fall into at least one of the following two categories: attempts to send inappropriate content to a web service, or attempts to breach security on a web service. Common types of attacks against XML-based web services include:

- **Malicious code or objects.** XML requests that contain code or objects that can either directly obtain sensitive information or can give an attacker control of the web service or underlying server.
- **Badly-formed XML requests.** XML requests that do not conform to the W3C XML specification, and that can therefore breach security on an insecure web service.
- **Denial of service (DoS) attacks.** XML requests that are sent repeatedly and in high volume, with the intent of overwhelming the targeted web service and denying legitimate users access to the web service.

In addition to standard XML-based attacks, XML web services and Web 2.0 sites are also vulnerable to SQL injection and cross-site scripting attacks, as described below:

- **SQL injection attacks.** Sending an active SQL command or commands in an XML-based request, with the goal of causing an SQL database to execute that command or commands. As with HTML SQL injection attacks, XML SQL injection attacks are normally used to obtain unauthorized information.
- **Cross-site scripting attacks.** Using a script included in an XML based application to violate the same-origin policy, which does not allow any script to obtain properties from or modify any content on a different application. Since scripts can obtain information and modify files by using your XML application, allowing a script access to content belonging to a different application can give an attacker the means to obtain unauthorized information, to compromise the

application, or both.

Known web attacks can usually be stopped by filtering web site traffic for specific characteristics (signatures) that always appear for a specific attack and should never appear in legitimate traffic. This approach has the advantages of requiring relatively few resources and posing relatively little risk of false positives. It is therefore a valuable tool in fighting attacks on web sites and web services, and configuring basic signature protections that intercept most known web attacks is easy to do.

The greatest threat against web sites and applications does not come from known attacks, but from unknown attacks. Most unknown attacks fall into one of two categories: newly-launched attacks for which security firms have not yet developed an effective defense (zero-day attacks), and carefully-targeted attacks on a specific web site or web service rather than many web sites or web services (spear attacks). These attacks, like known attacks, are usually intended to obtain sensitive private information, compromise the web site or web service and allow it to be used for further attacks, or both of those goals.

Zero-day attacks are a major threat to all users. These attacks are usually of the same types as known attacks; zero-day attacks often involve injected SQL, a cross-site script, a cross-site request forgery, or another type of attack similar to known attacks. In most cases, they target vulnerabilities that the developers of the targeted software, web site, or web service either are unaware of or have just learned about. Security firms have therefore usually not developed defenses against these attacks, and even if they have, users have usually not obtained and installed the patches or performed the workarounds necessary to protect against these attacks. The time between discovery of a zero-day attack and availability of a defense (the vulnerability window) is shrinking, but perpetrators can still count on hours or even days in which many web sites and web services lack any specific protection against the attack.

Spear attacks are a major threat, but to a more select group of users. A common type of spear attack, a spear phish, is usually targeted at customers of a specific bank or financial institution, or (less commonly) at employees of a specific company or organization. Unlike other phishes, which are often crudely written forgeries that a user with any familiarity with the actual communications of that bank or financial institution can recognize, spear phishes are letter perfect and extremely convincing. They can contain information specific to the individual that, at first look, no stranger should know or be able to obtain. The spear phisher is therefore able to convince his or her target to provide the requested information, which the phisher can then use to loot accounts, to process illegitimately obtained money from other sources, or to gain access to other, even more sensitive information.

Both of these types of attack have certain characteristics that can usually be detected, although not by using static patterns that look for specific characteristics, as do standard signatures. Detecting these types of attacks requires more sophisticated and more resource-intensive approaches, such as heuristic filtering and positive security model systems. Heuristic filtering looks, not for specific patterns, but for patterns of behaviors. Positive security model systems model the normal behavior of the web site or web service that they are protecting, and then block connections that do not fit within that model of normal use. URL based and web-form based security checks profile normal use of your web sites, and then control how users interact with your web sites, using both heuristics and positive security to block anomalous or unexpected traffic. Both heuristic and positive security, properly designed and deployed, can catch most attacks that signatures miss. However, they require considerably more resources than do signatures, and you must spend some time configuring them properly to avoid false positives. They are therefore usually used, not as the primary line of defense, but as backups to signatures or other less resource-intensive approaches.

By configuring these advanced protections in addition to signatures, you create a hybrid security model, which enables the application firewall to provide comprehensive protection against both known and unknown attacks.

How The Application Firewall Works

Sep 03, 2013

When you install the application firewall, you create an initial security configuration, which consists of a policy, a profile, and a signatures object. The policy is a rule that identifies the traffic to be filtered, and the profile identifies the patterns and types of behavior to allow or block when the traffic is filtered. The simplest patterns, which are called signatures, are not specified within the profile, but in a signatures object that is associated with the profile.

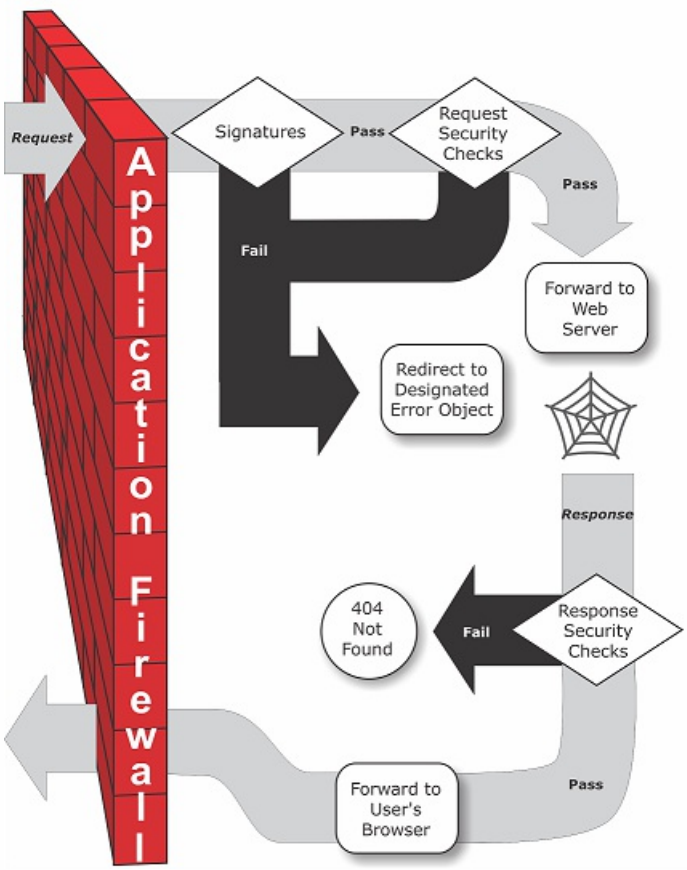
A signature is a string or pattern that matches a known type of attack. The application firewall contains over a thousand signatures in seven categories, each directed at attacks on specific types of web servers and web content. Citrix updates the list with new signatures as new threats are identified. During configuration, you specify the signature categories that are appropriate for the web servers and content that you need to protect. Signatures provide good basic protection with low processing overhead. If your applications have special vulnerabilities or you detect an attack against them for which no signature exists, you can add your own signatures.

The more advanced protections are called security checks. A security check is a more rigorous, algorithmic inspection of a request for specific patterns or types of behavior that might indicate an attack or constitute a threat to your protected web sites and web services. It can, for example, identify a request that attempts to perform a certain type of operation that might breach security, or a response that includes sensitive private information such as a social security number or credit card number. During configuration, you specify the security checks that are appropriate for the web servers and content that you need to protect. The security checks are restrictive. Many of them can block legitimate requests and responses if you do not add the appropriate exceptions (relaxations) when configuring them. Identifying the needed exceptions is not difficult if you use the adaptive learning feature, which observes normal use of your web site and creates recommended exceptions.

The application firewall can be installed as either a Layer 3 network device or a Layer 2 network bridge between your servers and your users, usually behind your company's router or firewall. It must be installed in a location where it can intercept traffic between the web servers that you want to protect and the hub or switch through which users access those web servers. You then configure the network to send requests to the application firewall instead of directly to your web servers, and responses to the application firewall instead of directly to your users. The application firewall filters that traffic before forwarding it to its final destination, using both its internal rule set and your additions and modifications. It blocks or renders harmless any activity that it detects as harmful, and then forwards the remaining traffic to the web server. The following figure provides an overview of the filtering process.

Note: The figure omits the application of a policy to incoming traffic. It illustrates a security configuration in which the policy is to process all requests. Also, in this configuration, a signatures object has been configured and associated with the profile, and security checks have been configured in the profile.

Figure 1. A Flowchart of Application Firewall Filtering



As the figure shows, when a user requests a URL on a protected web site, the application firewall first examines the request to ensure that it does not match a signature. If the request matches a signature, the application firewall either displays the error object (a web page that is located on the application firewall appliance and which you can configure by using the imports feature) or forwards the request to the designated error URL (the error page). Signatures do not require as many resources as do security checks, so detecting and stopping attacks that are detected by a signature before running any of the security checks reduces the load on the server.

If a request passes signature inspection, the application firewall applies the request security checks that have been enabled. The request security checks verify that the request is appropriate for your web site or web service and does not contain material that might pose a threat. For example, security checks examine the request for signs indicating that it might be of an unexpected type, request unexpected content, or contain unexpected and possibly malicious web form data, SQL commands, or scripts. If the request fails a security check, the application firewall either sanitizes the request and then sends it back to the NetScaler appliance (or NetScaler virtual appliance), or displays the error object. If the request passes the security checks, it is sent back to the NetScaler appliance, which completes any other processing and forwards the request to the protected web server.

When the web site or web service sends a response to the user, the application firewall applies the response security checks that have been enabled. The response security checks examine the response for leaks of sensitive private information, signs of web site defacement, or other content that should not be present. If the response fails a security check, the application firewall either removes the content that should not be present or blocks the response. If the response passes the security checks, it is sent back to the NetScaler appliance, which forwards it to the user.

Updated: 2013-09-03

The basic application firewall features are policies, profiles, and signatures, which provide a hybrid security model as described in "Known Web Attacks," "Unknown Web Attacks," and "How the Application Firewall Works." Of special note is the learning feature, which observes traffic to your protected applications and recommends appropriate configuration settings for certain security checks.

The imports feature manages files that you upload to the application firewall. These files are then used by the application firewall in various security checks, or when responding to a connection that matches a security check.

You can use the logs, statistics, and reports features to evaluate the performance of the application firewall and identify possible needs for additional protections.

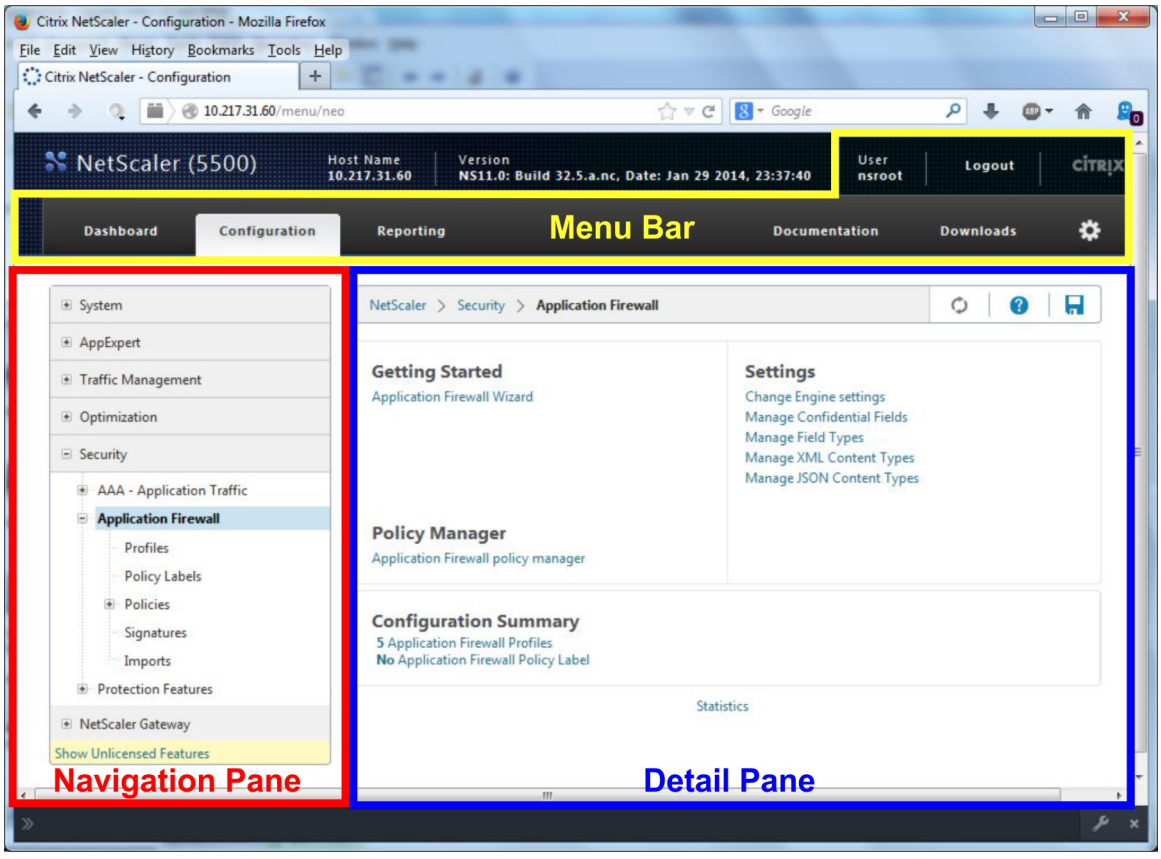
The Application Firewall Configuration Interfaces

Oct 09, 2014

All hardware and virtual versions of the Citrix NetScaler application delivery controller (ADC) can be configured and managed from the Citrix NetScaler command line interface or the web-based configuration utility. All features of most NetScaler features can be configured using either of these tools. The Citrix Application Firewall is an exception: not all application firewall configuration tasks can be performed at the command line. Inexperienced users also find the configuration utility easier to use. In particular, the application firewall wizard considerably reduces the complexity of configuring the application firewall. Unlike most NetScaler wizards, the application firewall wizard can serve as your primary interface to the application firewall.

The command line interface is a modified UNIX shell based on the FreeBSD bash shell. To configure the application firewall from the command line interface, you type commands at the prompt and press the Enter key, just as you do with any other Unix shell. For instructions for using the command line interface, see "Command Reference."

The configuration utility is a web-based GUI interface to the ADC. The application firewall configuration section is found under Security > Application Firewall. Figure 1 shows the navigation pane expanded to display the application firewall screens, and in the detail pane the main application firewall screen.



The configuration utility has two main areas on all screens. The panel on the left, called the navigation pane, contains a navigation tree, with which you navigate to the screens on which you configure the features that are installed on your appliance. The screens to which you navigate appear to the right of the navigation pane, in the details pane.

When you access the configuration utility, the details pane displays the System Overview screen. If, in the navigation pane,

you click plus sign next to the application firewall folder, the Application Firewall node expands to include the main application firewall elements that you can configure. If you click the first element, Profiles, the details pane displays the configured profiles, if any profiles have been configured. At the bottom of the details pane, you can click Add to configure a new profile. Other buttons at the bottom of the details pane are grayed out until you select an existing profile. Screens for the other elements work in the same way.

If, instead of expanding the application firewall node, you click the node itself, the details pane displays different options, one of which is the application firewall wizard, as shown in Figure 1. Citrix recommends that you use the wizard for initial configuration, and many users use it almost exclusively. It includes most of the functionality that is available elsewhere in the configuration utility.

For information and instructions on accessing the configuration utility, see "Citrix NetScaler Getting Started Guide."

Configuring the Application Firewall

Oct 09, 2014

You can configure the Citrix Application Firewall (application firewall) by using any of the following methods:

- **Application Firewall Wizard.** A dialog box consisting of a series of screens that step you through the configuration process.
- **Citrix Web Interface AppExpert Template.** A NetScaler AppExpert template (a set of configuration settings) that are designed to provide appropriate protection for web sites. This AppExpert template contains appropriate Application Firewall configuration settings for protecting many web sites.
- **Citrix NetScaler Configuration Utility.** The NetScaler web-based configuration interface.
- **Citrix NetScaler Command Line Interface.** The NetScaler command line configuration interface.

Citrix recommends that you use the Application Firewall Wizard. Most users will find it the easiest method to configure the application firewall, and it is designed to prevent mistakes. If you have a new Citrix NetScaler ADC or VPX that you will use primarily to protect web sites, you may find the Web Interface AppExpert template a better option because it provides a good default configuration, not just for the application firewall, but for the entire appliance. Both the configuration utility and the command line interface are intended for experienced users, primarily to modify an existing configuration or use advanced options.

The application firewall wizard is a dialog box that consists of several screens that prompt you to configure each part of a simple configuration. The application firewall then creates the appropriate configuration elements from the information that you give it. This is the simplest and, for most purposes, the best way to configure the application firewall.

To use the wizard, connect to the configuration utility with the browser of your choice. When the connection is established, verify that the application firewall is enabled, and then run the application firewall wizard, which prompts you for configuration information. You do not have to provide all of the requested information the first time you use the wizard. Instead, you can accept default settings, perform a few relatively straightforward configuration tasks to enable important features, and then allow the application firewall to collect important information to help you complete the configuration.

For example, when the wizard prompts you to specify a rule for selecting the traffic to be processed, you can accept the default, which selects all traffic. When it presents you with a list of signatures, you can enable the appropriate categories of signatures and turn on the collection of statistics for those signatures. For this initial configuration, you can skip the advanced protections (security checks). The wizard automatically creates the appropriate policy, signatures object, and profile (collectively, the security configuration) , and binds the policy to global. The application firewall then begins filtering connections to your protected web sites, logging any connections that match one or more of the signatures that you enabled, and collecting statistics about the connections that each signature matches. After the application firewall processes some traffic, you can run the wizard again and examine the logs and statistics to see if any of the signatures that you have enabled are matching legitimate traffic. After determining which signatures are identifying the traffic that you want to block, you can enable blocking for those signatures. If your web site or web service is not complex, does not use SQL, and does not have access to sensitive private information, this basic security configuration will probably provide adequate protection.

You may need additional protection if, for example, your web site is dynamic. Content that uses scripts may need protection against cross-site scripting attacks. Web content that uses SQL—such as shopping carts, many blogs, and most

content management systems—may need protection against SQL injection attacks. Web sites and web services that collect sensitive private information such as social security numbers or credit card numbers may require protection against unintentional exposure of that information. Certain types of web-server or XML-server software may require protection from types of attacks tailored to that software. Another consideration is that specific elements of your web sites or web services may require different protection than do other elements. Examining the application firewall logs and statistics can help you identify the additional protections that you might need.

After deciding which advanced protections are needed for your web sites and web services, you can run the wizard again to configure those protections. Certain security checks require that you enter exceptions (relaxations) to prevent the check from blocking legitimate traffic. You can do so manually, but it is usually easier to enable the adaptive learning feature and allow it to recommend the necessary relaxations. You can use the wizard as many times as necessary to enhance your basic security configuration and/or create additional security configurations.

The wizard automates some tasks that you would have to perform manually if you did not use the wizard. It automatically creates a policy, a signatures object, and a profile, and assigns them the name that you provided when you were prompted for the name of your configuration. The wizard also adds your advanced-protection settings to the profile, binds the signatures object to the profile, associates the profile with the policy, and puts the policy into effect by binding it to Global.

A few tasks cannot be performed in the wizard. You cannot use the wizard to bind a policy to a bind point other than Global. If you want the profile to apply to only a specific part of your configuration, you must manually configure the binding. You cannot configure the engine settings or certain other global configuration options in the wizard. While you can configure any of the advanced protection settings in the wizard, if you want to modify a specific setting in a single security check, it may be easier to do so on the manual configuration screens in the configuration utility.

For more information on using the Application Firewall Wizard, see "The Application Firewall Wizard."

AppExpert Templates are a different and simpler approach to configuring and managing complex enterprise applications. The AppExpert display in the configuration utility consists of a table. Applications are listed in the left-most column, with the NetScaler features that are applicable to that application appearing each in its own column to the right. (In the AppExpert interface, those features that are associated with an application are called *application units*.) In the AppExpert interface, you configure the interesting traffic for each application, and turn on rules for compression, caching, rewrite, filtering, responder and the application firewall, instead of having to configure each feature individually.

The Web Interface AppExpert Template contains rules for the following application firewall signatures and security checks:

- "Deny URL check." Detects connections to content that is known to pose a security risk, or to any other URLs that you designate.
- "Buffer Overflow check." Detects attempts to cause a buffer overflow on a protected web server.
- "Cookie Consistency check." Detects malicious modifications to cookies set by a protected web site.
- "Form Field Consistency check." Detects modifications to the structure of a web form on a protected web site.
- "CSRF Form Tagging check." Detects cross-site request forgery attacks.
- "Field Formats check." Detects inappropriate information uploaded in web forms on a protected web site.
- "HTML SQL Injection check." Detects attempts to inject unauthorized SQL code.
- "HTML Cross-Site Scripting check." Detects cross-site scripting attacks.

For information on installing and using an AppExpert Template, see "AppExpert Applications and Templates."

The NetScaler configuration utility is a web-based interface that provides access to all configuration options for the application firewall feature, including advanced configuration and management options that are not available from any other configuration tool or interface. Specifically, many advanced Signatures options can be configured only in the configuration utility. You can review recommendations generated by the learning feature only in the configuration utility. You can bind policies to a bind point other than Global only in the configuration utility.

For a description of the configuration utility, see "The Application Firewall Configuration Interfaces." For more information on using the configuration utility to configure the application firewall, see "Manual Configuration By Using the Configuration Utility."

For instructions on configuring the application firewall by using the configuration utility, see "Manual Configuration By Using the Configuration Utility." For information on the Citrix NetScaler Configuration Utility, see "The Application Firewall Configuration Interfaces."

The Citrix NetScaler command line interface is a modified UNIX shell based on the FreeBSD bash shell. To configure the Application Firewall from the command line interface, you type commands at the prompt and press the Enter key, just as you do with any other Unix shell. You can configure most parameters and options for the application firewall by using the NetScaler command line. Exceptions are the signatures feature, many of whose options can be configured only by using the configuration utility or the Application Firewall wizard, and the learning feature, whose recommendations can only be reviewed in the configuration utility.

For instructions on configuring the application firewall by using the NetScaler command line, see "Manual Configuration By Using the Command Line Interface."

Enabling the Application Firewall

Oct 09, 2014

Before you can create an application firewall security configuration, you must make sure that the application firewall feature is enabled.

- If you are configuring a dedicated Citrix Application Firewall ADC or are upgrading an existing Citrix NetScaler ADC or VPX, the feature is already enabled. You do not have to perform either of the procedures described here.
- If you have a new NetScaler ADC or VPX, you need to enable the application firewall feature before you configure it.
- If you are upgrading a NetScaler ADC or VPX from a previous version of the NetScaler operating system to the current version, you might need to enable the application firewall feature before you configure it.

Note: If you are upgrading a NetScaler ADC or VPX from a previous version, you might also need to update the licenses on your ADC or VPX before you can enable the application firewall. Check with your Citrix representative or reseller to obtain the correct license.

You can enable the application firewall by using the command line or the configuration utility.

At the command prompt, type the following command:

enable ns feature AppFW

1. Navigate to System > Settings.
2. In the details pane, click Configure Basic Features.
3. In the Configure Basic Features dialog box, check the Application Firewall check box.
4. Click OK.

The Application Firewall Wizard

Oct 01, 2013

Unlike most wizards, the Application Firewall wizard is designed not just to simplify the initial configuration process, but also to modify previously created configurations and to maintain your Application Firewall setup. A typical user runs the wizard multiple times, skipping some of the screens each time.

To run the Application Firewall wizard, first open the configuration utility. Next, in the navigation pane, expand Application Firewall, and then in the details pane click Application Firewall Wizard. (For more information about the configuration utility, see "The Application Firewall User Interfaces.") Then:

1. Navigate to Security > Application Firewall.
2. In the details pane, under Getting Started, click Application Firewall Wizard. The first screen of the wizard appears.
3. To advance to the next screen, click Next.

The Application Firewall wizard displays the following screens, in the following order:

1. **Introduction screen.** Provides an introduction to the Application Firewall wizard. There is nothing that you can configure on this screen.

2. **Specify Name screen.** On this screen, when creating a new security configuration, you specify the name that the wizard is to assign to the configuration. The name can begin with a letter, number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) symbols. Choose a name that makes it easy for others to tell what content your new security configuration protects.

   Note: Because the wizard uses this name for both the policy and the profile, it is limited to 31 characters. Manually created policies can have names up to 127 characters in length.
   When creating an existing configuration, you select Modify Existing Configuration and then, in the Name drop-down list, select the name of the existing configuration that you want to modify.

   Note: Only policies that are bound to global or to a bind point appear in this list; you cannot modify an unbound policy by using the Application Firewall wizard. You must either manually bind it to Global or a bind point, or modify it manually. (For manual modification, in the configuration utility's Application Firewall --> Policies --> Firewall pane, select the policy and click Open).
   You also select a profile type on this screen. The profile type determines the types of advanced protection (security checks) that can be configured. Because certain kinds of content are not vulnerable to certain types of security threats, restricting the list of available checks saves time during configuration. The types of Application Firewall profiles are:

   - **Web Application (HTML).** Any HTML-based Web site that does not use XML or Web 2.0 technologies.
   - **XML Application (XML, SOAP).** Any XML-based Web service.
   - **Web 2.0 Application (HTML, XML, REST).** Any Web 2.0 site that combines HTML and XML-based content, such as an ATOM-based site, a blog, an RSS feed, or a wiki.

   Note: If you are unsure which type of content is used on your Web site, you can choose Web 2.0 Application to ensure that you protect all types of Web application content.

3. **Specify Rule screen.** On this screen, you specify the policy rule (*expression*) that defines the traffic to be examined by this security configuration. If you are creating an initial configuration to protect your Web sites and Web services, you can simply accept the default value, **true**, which selects all web traffic .

   If you want this security configuration to examine, not all HTTP traffic that is routed through the appliance, but specific traffic, you can write a policy rule specifying the traffic that you want it to examine. Rules are written in Citrix NetScaler expressions language, which is a fully functional object-oriented programming language.

   - For a simple description of using the NetScaler expressions syntax to create Application Firewall rules, and a list of useful rules, see "Firewall Policies."
   - For a detailed explanation of how to create policy rules in NetScaler expressions syntax, see "Policies and Expressions."

   Note: In addition to the default expressions syntax, for backward compatibility the NetScaler operating system supports the NetScaler classic expressions syntax on NetScaler Classic and nCore appliances and virtual appliances. Classic expressions are not supported on NetScaler Cluster appliances and virtual appliances. Current users who want to migrate their existing configurations to the NetScaler cluster must migrate any policies that contain classic expressions to the default expressions syntax.
4. **Select Signature Protections screen.** On this screen, you select the categories of signatures that you want to use to protect your web sites and web services. The default categories are:

   - **CGI.** Protection against attacks on web sites that use CGI scripts in any language, including PERL scripts, Unix shell scripts, and Python scripts.
   - **Cold Fusion.** Protection against attacks on web sites that use the Adobe Systems® ColdFusion® Web development platform.
   - **FrontPage.** Protection against attacks on web sites that use the Microsoft® FrontPage® Web development platform.
   - **PHP.** Protection against attacks on web sites that use the PHP open-source Web development scripting language.
   - **Client side.** Protection against attacks on client-side tools used to access your protected web sites, such as Microsoft Internet Explorer, Mozilla Firefox, the Opera browser, and the Adobe Acrobat Reader.
   - **Microsoft IIS.** Protection against attacks on Web sites that run the Microsoft Internet Information Server (IIS).
   - **Miscellaneous.** Protection against attacks on other server-side tools, such as Web servers and database servers.

   If you are creating a new security configuration, the signature categories that you select are enabled, and by default they are recorded in a new signatures object. The new signatures object is assigned the same name that you entered on the Specify name screen as the name of the security configuration.

   If you have previously configured signatures objects and want to use one of them as the signatures object associated with the security configuration that you are creating, click Select Existing Signature and select a signatures object from the Signatures list.

   If you are modifying an existing security configuration, you can click Select Existing Signature and assign a different signatures object to the security configuration.

5. **Select Signature Actions screen.** On this screen, you select the actions associated with the signature categories that you selected on the Select signature protections screen. If you are creating an initial configuration, you might want

to accept the defaults, which enable the Log and Stats actions but not the Block action. You can decide later, after reviewing the collected logs and statistics, which signatures you should use to block traffic, and then enable the Block action for those signatures. Signatures are designed to catch specific known attacks on your web sites, and therefore they have extremely low false positive rates. However, with any new configuration, you should probably observe how the settings you chose are working before you use them to block traffic.

If you select More for one of the signature categories, the Configure Actions for Signatures dialog box appears. Its contents are the same as the contents of the Modify Signatures Object dialog box, as described in "To Configure a Signatures Object."

If the signatures object has already logged connections, you can click Logs to display the Syslog Viewer with the logs, as described in "Logs, Statistics, and Reports." If a signature is blocking legitimate access to your protected web site or web service, you can create and implement a relaxation for that signature by selecting a log that shows the unwanted blocking, and then clicking Deploy.

6. **Select Advanced Protections screen.** On this screen, you choose the advanced protections (also called *security checks* or simply *checks*) that you want to use to protect your web sites and web services. The checks are divided into categories. Which categories are available (and which checks are available within a category) depends on the profile type that you chose on the Specify Name screen. All checks are available for Web 2.0 Application profiles. If you chose that profile type, the Select advanced protections screen displays the following categories of security checks:
   - Top--level protections (Some checks appear at the top level, not in any category.)
   - Data Leak Prevention Protections
   - Advanced Form Protections
   - URL Protections
   - XML Protections

   To display the individual checks in a category, click the icon to the left of the category. To apply a security check to your filtered data, select the check box next to the name of the security check. For descriptions of the security checks see "Advanced Protections" and its subtopics.
7. **Select Advanced Actions screen.** On this screen, you configure the actions for the advanced protections that you have enabled.
   Note: If no advanced protections are enabled, the Wizard skips the Advanced Actions screen and goes directly to the Summary screen.
   The actions that you can configure are:
   - **Block.** Block connections that match the signature. Disabled by default.
   - **Log.** Log connections that match the signature for later analysis. Enabled by default.
   - **Stats.** Maintain statistics, for each signature, that show how many connections it matched and provide certain other information about the types of connections that were blocked. Disabled by default.
   - **Learn.** Observe traffic to this Web site or Web service, and use connections that repeatedly violate this check to generate recommended exceptions to the check, or new rules for the check. Available only for some checks.

   To enable or disable an action for a check, in the list, select or clear the check box for that action to the right of that check.

   To configure other parameters for those checks that have them, in the list, click the blue chevron to the far right of that check. In the dialog box that appears, configure the parameters. These vary from check to check. You can also select a check and, at the bottom of the dialog box, click Open to display a dialog box for modifying any of the options for that check. These dialog boxes also vary from check to check. Most of them include a Checks tab and a General tab. If the check supports relaxations, the Checks tab includes an Add button, which opens yet another dialog box, in which you can specify a relaxation for the check. A relaxation is a rule for exempting specified traffic from the check.

For information about the settings available for a check, see the detailed description of that check.

To review the recommendations generated by the learning engine for a specific check, select that check and then click Learned Violations to open the Manage Learned Rules dialog box for that check. For more information on how learning works and how to configure exceptions (relaxations) or deploy learned rules for a check, see "Manual Configuration By Using the Configuration Utility" under To configure and use the learning feature

To view all logs for a specific check, select that check, and then click Logs to display the Syslog Viewer, as described in "Logs, Statistics, and Reports." If a security check is blocking legitimate access to your protected web site or web service, you can create and implement a relaxation for that security check by selecting a log that shows the unwanted blocking, and then clicking Deploy.

8. **Summary screen.** On this screen, you review your configuration choices to verify that they are what you want. If you want to make changes, you click Back until you have returned to the appropriate screen, and make your changes. If the configuration is as you want it, you click Finish to save it , and then click Exit to close the Application Firewall wizard.

Following are four procedures that show how to perform specific types of configuration by using the Application Firewall wizard.

1. Navigate to Security > Application Firewall.
2. In the details pane, under Getting Started, click Application Firewall Wizard.
3. On the Application Firewall wizard, Introduction screen, in the lower right-hand corner, click Next.
4. On the Specify Name screen, in the Name text box, type a name for your new security configuration, and from the Type drop-down list, select the type of security configuration. Then, click Next.
5. On the Specify Rule screen, click Next again.
   Note: The default rule, **true**, protects all Web traffic that is sent via your NetScaler appliance or virtual appliances. You can create specific security configurations to protect specific parts of your Web sites or Web applications later.
6. On the Select Signature Protections screen, select check boxes to specify the groups of signatures that are appropriate for protecting the content on your protected web sites, and then click Next.
   For more information about signatures, see "Signatures."
7. On the Select Signature Actions screen, select or clear the associated check boxes to choose the signature actions that you want for each signature category that you selected in the previous step, and then click Next.
8. On the Select Advanced Protections screen, click Next again.
   You typically do not need to configure the security checks during initial configuration.
9. On the Summary screen, review your choices to verify that they are what you want. Then, click Finish, or click Back to return to a previous screen and make changes. When you are finished, click Exit to close the Application Firewall wizard.

1. Navigate to Security > Application Firewall.
2. In the details pane, under Getting Started, click Application Firewall Wizard.
3. On the Application Firewall wizard, Introduction screen, in the lower right-hand corner, click Next.
4. On the Specify Name screen, select Modify Existing Configuration and, in the Name drop-down list, choose the security configuration that you created during simple configuration, and then click Next.
5. In the Specify Rule screen, click Next again.
6. In the Select Signature Protections screen, click Next again.

7. In the Select Signature Actions screen, enable blocking for your chosen signatures by selecting the Block check box to the left of each of those signature.
   For more information about which signatures to consider for blocking and how to determine when you can safely enable blocking for a signature, see "Signatures."

8. In the Select advanced protections screen, click Next.
9. On the Summary screen, review your choices to verify that they appear correct. Then, click Finish, or click Back to return to the Select Signature Actions screen and make changes. When you are finished, click Exit to close the Application Firewall wizard.

1. Navigate to Security > Application Firewall.
2. In the details pane, under Getting Started, click Application Firewall Wizard.
3. On the Application Firewall wizard, Introduction screen, in the lower right-hand corner, click Next.
4. On the Specify Name screen, select Modify Existing Configuration and, in the Name drop-down list, choose the security configuration that you created during simple configuration. Then, click Next.
5. On the Specify Rule screen, click Next again.
6. On the Select Signature Protections screen, click Next.
7. On the Select Signature Actions screen, click Next again.
8. On the Select advanced protections screen, select the check box beside each security check that you want to enable, and then click Next.
   For information about the security checks, see "Advanced Protections" and its subtopics.

9. On the Select Deep Actions screen, select check boxes to specify the actions that you want the Application Firewall to perform for each security check, and then click Next.
   For general information about the actions, see "Advanced Protections" and its subtopics. For information about the learning feature, which is available for some security checks, see "To configure and use the Learning feature."

10. On the Summary screen, review your choices to verify that they appear correct. Then, click Finish, or click Back to return to the Select Signature Actions screen and make changes. When you are finished, click Exit to close the Application Firewall wizard.

The following procedure describes how to use the Application Firewall wizard to create a specialized security configuration to protect only specific content. In this case, you create a new security configuration instead of modifying the initial configuration. This type of security configuration requires a custom rule, so that the policy applies the configuration to only the selected Web traffic.

1. Navigate to Security > Application Firewall.
2. In the details pane, under Getting Started, click Application Firewall Wizard.
3. On the Application Firewall wizard, Introduction screen, in the lower right-hand corner, click Next.
4. On the Specify Name screen, type a name for your new security configuration in the Name text box, select the type of security configuration from the Type drop-down list, and then click Next.
5. On the Specify Rule screen, enter a rule that matches only that content that you want this Web application to protect, and then click Next.
   For a description of policies and policy rules, see "Policies."

6. On the Select Signature Protections screen, choose the appropriate groups of signatures to protect the content on

your protected web sites by selecting the check box beside each group of signatures, and then click Next.
For detailed information about signatures, see "Signatures."

7. On the Select Signature Actions screen, select or clear the associated check boxes to choose the signature actions that you want for each signature category that you selected in the previous step, and then click Next. For a detailed description of actions, see "Signatures."
8. In the Select Advanced Protections screen, select the check box beside each security check that you want to enable, and then click Next.
   For detailed information about the security checks, see "Advanced Protections" and its subtopics.
9. In the Select Advanced Actions screen, select check boxes to specify the actions that you want the Application Firewall to perform for each security check. Then, click Next.
   For information about each security check to help you determine which actions to enable, see the Advanced Protections section.
10. On the Summary screen, review your choices to verify that they appear correct. Then, click Finish, or click Back to return to the Select Signature Actions screen and make changes. When you are finished, click Exit to close the wizard.

Manual Configuration

Mar 28, 2012

If you want to bind a profile to a bind point other than Global, you must manually configure the binding. Also, certain security checks require that you either manually enter the necessary exceptions or enable the learning feature to generate the exceptions that your Web sites and Web services need. Some of these tasks cannot be performed by using the application firewall wizard.

If you are familiar with how the application firewall works and prefer manual configuration, you can manually configure a signatures object and a profile, associate the signatures object with the profile, create a policy with a rule that matches the web traffic that you want to configure, and associate the policy with the profile. You then bind the policy to Global, or to a bind point, to put it into effect, and you have created a complete security configuration.

For manual configuration, you can use the configuration utility (a graphical interface) or the command line. Citrix recommends that you use the configuration utility. Not all configuration tasks can be performed at the command line. Certain tasks, such as enabling signatures and reviewing learned data, must be done in the configuration utility. Most other tasks are easier to perform in the configuration utility.

Manual Configuration By Using the Configuration Utility

Sep 03, 2013

If you need to configure the Application Firewall feature manually, Citrix recommends that you use the configuration utility. For a description of the configuration utility, see "The Application Firewall User Interfaces."

Before you can configure the signatures, you must create a new signatures object from the appropriate default signatures object template. Assign the copy a new name, and then configure the copy. You cannot configure or modify the default signatures objects directly. The following procedure provides basic instructions for configuring a signatures object. For more detailed instructions, see "Manually Configuring the Signatures Feature." If you need to create your own, user defined signatures, see "The Signatures Editor."

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to use as a template, and then click Add.
   Your choices are:
   - *** Default Signatures.** Contains the signatures rules, the SQL injection rules, and the cross-site scripting rules.
   - *** XPath Injection.** Contains all of the items in the * Default Signatures, and in addition contains the XPath injection rules.
3. In the Add Signatures Object dialog box, type a name for your new signatures object, click OK, and then click Close. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), and underscore (_) symbols.
4. Select the signatures object that you created, and then click Open.
5. In the Modify Signatures Object dialog box, set the Display Filter Criteria options at the left to display the filter items that you want to configure.
   As you modify these options, the results that you specify are displayed in the Filtered Results window at the right. For more information about the categories of signatures, see "Signatures."
6. In the Filtered Results area, configure the settings for a signature by selecting and clearing the appropriate check boxes.
7. When finished, finished, click Close.

Creating an application firewall profile requires that you specify only a few configuration details.

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, click Add.
3. In the Create Application Firewall Profile dialog box, type a name for your profile.
   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) symbols.
4. Choose the profile type from the drop-down list.
5. Click Create, and then click Close.

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, select the profile that you want to configure, and then click Open.
3. In the Configure Application Firewall Profile dialog box, on the Security Checks tab, configure the security checks.
   - To enable or disable an action for a check, in the list, select or clear the check box for that action.
   - To configure other parameters for those checks that have them, in the list, click the blue chevron to the far right of that check. In the dialog box that appears, configure the parameters. These vary from check to check.
     You can also select a check and, at the bottom of the dialog box, click Open to display the Configure Relaxation dialog box or Configure Rule dialog box for that check. These dialog boxes also vary from check to check. Most of them include a Checks tab and a General tab. If the check supports relaxations or user-defined rules, the Checks tab includes an Add button, which opens yet another dialog box, in which you can specify a relaxation or rule for the check. (A relaxation is a rule for exempting specified traffic from the check.) If relaxations have already been configured, you can select one and click Open to modify it.
   - To review learned exceptions or rules for a check, select the check, and then click Learned Violations. In the Manage Learned Rules dialog box, select each learned exception or rule in turn.
     - To edit the exception or rule, and then add it to the list, click Edit & Deploy.
     - To accept the exception or rule without modification, click Deploy.
     - To remove the exception or rule from the list, click Skip.
   - To refresh the list of exceptions or rules to be reviewed, click Refresh.
   - To open the Learning Visualizer and use it to review learned rules, click Visualizer.
   - To review the log entries for connections that matched a check, select the check, and then click Logs. You can use this information to determine which checks are matching attacks, so that you can enable blocking for those checks. You can also use this information to determine which checks are matching legitimate traffic, so that you can configure an appropriate exemption to allow those legitimate connections. For more information about the logs, see "Logs, Statistics, and Reports."
   - To completely disable a check, in the list, clear all of the check boxes to the right of that check.
4. On the Settings tab, configure the profile settings.
   - To associate the profile with the set of signatures that you previously created and configured, under Common Settings, choose that set of signatures in the Signatures drop-down list.
     Note: You may need to use the scroll bar on the right of the dialog box to scroll down to display the Common Settings section.
   - To configure an HTML or XML Error Object, select the object from the appropriate drop-down list.
     Note: You must first upload the error object that you want to use in the Imports pane. For more information about importing error objects, see "Imports."
   - To configure the default XML Content Type, type the content type string directly into the Default Request and Default Response text boxes, or click Manage Allowed Content Types to manage the list of allowed content types. ">>More...."
5. If you want to use the learning feature, click Learning, and configure the learning settings for the profile, as described in "Configuring and Using the Learning Feature".
6. Click OK to save your changes and return to the Profiles pane.

Updated: 2014-06-12

You configure two different types of information in this dialog box, depending upon which security check you are configuring. In the majority of cases, you configure an exception (or relaxation) to the security check. If you are configuring the Deny URL check or the Field Formats check, you configure an addition (or rule). The process for either of these is the

same.

To configure a relaxation or rule by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the Profiles pane, select the profile you want to configure, and then click Open.
3. In the Configure Application Firewall Profile dialog box, click the Security Checks tab. The Security Checks tab contains the complete list of application firewall security checks, also called *advanced protections* in some places.
4. In the Security Checks tab, click the check that you want to configure, and then click Open. The Modify Check dialog box for the check that you chose is displayed, with the Checks tab selected. The Checks tab contains a list of existing relaxations or rules for this check. The list might be empty if you have not either manually added any relaxations or approved any relaxations that were recommended by the learning engine. Beneath the list is a row of buttons that allow you to add, modify, delete, enable, or disable the relaxations on the list.
5. To add or modify a relaxation or a rule, do one of the following:
   - To add a new relaxation, click Add.
   - To modify an existing relaxation, select the relaxation that you want to modify, and then click Open.

   The Add Check Relaxation or Modify Check Relaxation dialog box for the selected check is displayed. Except for the title, these dialog boxes are identical.
6. Fill in the dialog box as described below. The dialog boxes for each check are different; the list below covers all elements that might appear in any dialog box.
   - **Enabled check box**—Select to place this relaxation or rule in active use; clear to deactivate it.
   - **Attachment Content Type**—The Content-Type attribute of an XML attachment. In the text area, enter a regular expression that matches the Content-Type attribute of the XML attachments to allow.
   - **Action URL**—In the text area, enter a PCRE-format regular expression that defines the URL to which data entered into the web form is delivered.
   - **Cookie**—In the text area, enter a PCRE-format regular expression that defines the cookie.
   - **Field Name**—A web form field name element may be labeled Field Name, Form Field, or another similar name. In the text area, enter a PCRE-format regular expression that defines the name of the form field.
   - **Form Origin URL**—In the text area, enter a PCRE-format regular expression that defines the URL that hosts the web form.
   - **Form Action URL**—In the text area, enter a PCRE-format regular expression that defines the URL to which data entered into the web form is delivered.
   - **Name**—An XML element or attribute name. In the text area, enter a PCRE-format regular expression that defines the name of the element or attribute.
   - **URL**—A URL element may be labeled Action URL, Deny URL, Form Action URL, Form Origin URL, Start URL, or simply URL. In the text area, enter a PCRE-format regular expression that defines the URL.
   - **Format**—The format section contains multiple settings that include list boxes and text boxes. Any of the following can appear:
     - **Type**—Select a field type in the Type drop-down list. To add a new field type definition, click Manage—
     - **Minimum Length**—Type a positive integer that represents the minimum length in characters if you want to force

users to fill in this field. Default: 0 (Allows field to be left blank.)
    - **Maximum length**—To limit the length of data in this field, type a positive integer that represents the maximum length in characters. Default: 65535
- **Location**—Choose the element of the request that your relaxation will apply to from the drop-down list. For HTML security checks, the choices are:
    - FORMFIELD—Form fields in web forms.
    - HEADER—Request headers.
    - COOKIE—Set-Cookie headers.

  For XML security checks, the choices are:
    - ELEMENT—XML element.
    - ATTRIBUE—XML attribute.
- **Maximum Attachment Size**—The maximum size in bytes allowed for an XML attachment.
- **Comments**—In the text area, type a comment. Optional.

  Note: For any element that requires a regular expression, you can type the regular expression, use the Regex Tokens menu to insert regular expression elements and symbols directly into the text box, or click Regex Editor to open the Add Regular Expression dialog box, and use it to construct the expression.

7. To remove a relaxation or rule, select it, and then click Remove.
8. To enable a relaxation or rule, select it, and then click Enable.
9. To disable a relaxation or rule, select it, and then click Disable.
10. To configure the settings and relationships of all existing relaxations in an integrated interactive graphic display, click Visualizer, and use the display tools.

    Note: The Visualizer button does not appear on all check relaxation dialog boxes.
11. To review learned rules for this check, click Learning and perform the steps in "To configure and use the Learning feature."
12. Click OK.

1. Navigate to Security > Application Firewall > Profiles.
2. In the Profiles pane, select the profile, and then click Open.
3. Click the Learning tab. At the top of the Learning tab is list of the security checks that are available in the current profile and that support the learning feature.
4. To configure the learning thresholds, select a security check, and then type the appropriate values in the following text boxes:
    - **Minimum number threshold.** Depending on which security check's learning settings you are configuring, the minimum number threshold might refer to the minimum number of total user sessions that must be observed, the minimum number of requests that must be observed, or the minimum number of times a specific form field must be observed, before a learned relaxation is generated. Default: 1
    - **Percentage of times threshold.** Depending on which security check's learning settings you are configuring, the percentage of times threshold might refer to the percentage of total observed user sessions that violated the security check, the percentage of requests, or the percentage of times a form field matched a particular field type, before a learned relaxation is generated. Default: 0
5. To remove all learned data and reset the learning feature, so that it must start its observations again from the beginning, click Remove All Learned Data.

Note: This button removes only learned recommendations that have not been reviewed and either approved or skipped. It does not remove learned relaxations that have been accepted and deployed.

6. To restrict the learning engine to traffic from a specific set of IPs, click Trusted Learning Clients, and add the IP addresses that you want to use to the list.
   1. To add an IP address or IP address range to the Trusted Learning Clients list, click Add.
   2. In the Add Trusted Learning Clients dialog box, Trusted Clients IP list box, type the IP address or an IP address range in CIDR format.
   3. In the Comments text area, type a comment that describes this IP address or range.
   4. Click Create to add your new IP address or range to the list.
   5. To modify an existing IP address or range, click the IP address or range, and then click Open. Except for the name, the dialog box that appears is identical to the Add Trusted Learning Clients dialog box.
   6. To disable or enable an IP address or range, but leave it on the list, click the IP address or range, and then click Disable or Enable, as appropriate.
   7. To remove an IP address or range completely, click the IP address or range, and then click Remove.
7. Click Close to return to the Configure Application Firewall Profile dialog box.
8. Click Close to close the Configure Application Firewall Profile dialog box, and return to the Application Firewall Profile screen.

1. Navigate to Security > Application Firewall > Policies.
2. In the details pane, do one of the following:
   - To create a new firewall policy, click Add. The Create Application Firewall Policy is displayed.
   - To edit an existing firewall policy, select the policy, and then click Open.

   The Create Application Firewall Policy or Configure Application Firewall Policy is displayed.
3. If you are creating a new firewall policy, in the Create Application Firewall Policy dialog box, Policy Name text box, type a name for your new policy.
   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 128 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) symbols.

   If you are configuring an existing firewall policy, this field is read-only. You cannot modify it.
4. Select the profile that you want to associate with this policy from the Profile drop-down list. You can create a new profile to associate with your policy by clicking New, and you can modify an existing profile by clicking Modify.
5. In the Expression text area, create a rule for your policy.
   - You can type a rule directly into the text area.
   - You can click Prefix to select the first term for your rule, and follow the prompts. See "To Create an Application Firewall Rule (Expression)" for a complete description of this process.
   - You can click Add to open the Add Expression dialog box, and use it to construct the rule. See "The Add Expression Dialog Box" for a complete description of this process.
6. Click Create or OK, and then click Close.

The policy rule, also called the *expression*, defines the web traffic that the application firewall filters by using the profile associated with the policy. Like other NetScaler policy rules (or *expressions*), application firewall rules use NetScaler expressions syntax. This syntax is powerful, flexible, and extensible. It is too complex to describe completely in this set of instructions. You can use the following procedure to create a simple firewall policy rule, or you can read it as an overview of the policy creation process.

1. If you have not already done so, navigate to the appropriate location in the Application Firewall wizard or the NetScaler configuration utility to create your policy rule:
   - If you are configuring a policy in the Application Firewall wizard, in the navigation pane, click Application Firewall, then in the details pane click Application Firewall Wizard, and then navigate to the Specify Rule screen.
   - If you are configuring a policy manually, in the navigation pane, expand Application Firewall, then Policies, and then Firewall. In the details pane, to create a new policy, click Add. To modify an existing policy, select the policy, and then click Open.
2. On the Specify Rule screen, the Create Application Firewall Profile dialog box, or the Configure Application Firewall Profile dialog box, click Prefix, and then choose the prefix for your expression from the drop-down list. Your choices are:
   - **HTTP.** The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.
   - **SYS.** The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   - **CLIENT.** The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.
   - **SERVER.** The computer to which the request was sent. Choose this if you want to examine some aspect of the recipient of the request.

   After you choose a prefix, the application firewall displays a two-part prompt window that displays the possible next choices at the top, and a brief explanation of what the selected choice means at the bottom.
3. Choose your next term.
   If you chose HTTP as your prefix, your only choice is REQ, which specifies the Request/Response pair. (The application firewall operates on the request and response as a unit instead of on each separately.) If you chose another prefix, your choices are more varied. For help on a specific choice, click that choice once to display information about it in the lower prompt window.

   When you have decided which term you want, double-click it to insert it into the Expression window.
4. Type a period after the term you just chose. You are then prompted to choose your next term, as described in the previous step. When a term requires that you type a value, fill in the appropriate value. For example, if you choose HTTP.REQ.HEADER(""), type the header name between the quotation marks.
5. Continue choosing terms from the prompts and filling in any values that are needed, until your expression is finished. Following are some examples of expressions for specific purposes.

   - **Specific web host.** To match traffic from a particular web host:

     HTTP.REQ.HEADER("Host").EQ("shopping.example.com")
     For shopping.example.com, substitute the name of the web host that you want to match.

   - **Specific web folder or directory.** To match traffic from a particular folder or directory on a Web host:

     HTTP.REQ.URL.STARTSWITH("https//www.example.com/folder")
     For www.example.com, substitute the name of the web host. For folder, substitute the folder or path to the content that you want to match. For example, if your shopping cart is in a folder called /solutions/orders, you substitute that string for folder.

   - **Specific type of content: GIF images.** To match GIF format images:

     HTTP.REQ.URL.ENDSWITH(".gif")
     To match other format images, substitute another string in place of .gif.

- **Specific type of content: scripts.** To match all CGI scripts located in the CGI-BIN directory:

  HTTP.REQ.URL.STARTSWITH("https//www.example.com/CGI-BIN")
  To match all JavaScripts with .js extensions:

  HTTP.REQ.URL.ENDSWITH(".js")
  For more information about creating policy expressions, see "Policies and Expressions."

Note: If you use the command line to configure a policy, remember to escape any double quotation marks within NetScaler expressions. For example, the following expression is correct if entered in the configuration utility:
HTTP.REQ.HEADER("Host").EQ("shopping.example.com")
If entered at the command line, however, you must type this instead:

HTTP.REQ.HEADER(\"Host\").EQ(\"shopping.example.com\")

The Add Expression dialog box (also referred to as the Expression Editor) helps users who are not familiar with the NetScaler expressions language to construct a policy that matches the traffic that they want to filter.

1. If you have not already done so, navigate to the appropriate location in the Application Firewall wizard or the NetScaler configuration utility:
   - If you are configuring a policy in the Application Firewall wizard, in the navigation pane, click Application Firewall, then in the details pane click Application Firewall Wizard, and then navigate to the Specify Rule screen.
   - If you are configuring a policy manually, in the navigation pane, expand Application Firewall, then Policies, and then Firewall. In the details pane, to create a new policy, click Add. To modify an existing policy, select the policy, and then click Open.
2. On the Specify Rule screen, in the Create Application Firewall Profile dialog box, or in the Configure Application Firewall Profile dialog box, click Add.
3. In the Add Expression dialog box, in the Construct Expression area, in the first list box, choose one of the following prefixes:
   - **HTTP.** The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol. The default choice.
   - **SYS.** The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   - **CLIENT.** The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.
   - **SERVER.** The computer to which the request was sent. Choose this if you want to examine some aspect of the recipient of the request.
4. In the second list box, choose your next term. The available terms differ depending on the choice you made in the previous step, because the dialog box automatically adjusts the list to contain only those terms that are valid for the context. For example, if you selected HTTP in the previous list box, the only choice is REQ, for requests. Because the application firewall treats requests and associated responses as a single unit and filters both, you do not need to specific responses separately. After you choose your second term, a third list box appears to the right of the second. The Help window displays a description of the second term, and the Preview Expression window displays your expression.
5. In the third list box, choose the next term. A new list box appears to the right, and the Help window changes to display a description of the new term. The Preview Expression window updates to display the expression as you have specified it to that point.

6. Continue choosing terms, and when prompted filling in arguments, until your expression is complete. If you make a mistake or want to change your expression after you have already selected a term, you can simply choose another term. The expression is modified, and any arguments or additional terms that you added after the term that you modified are cleared.
7. When you have finished constructing your expression, click OK to close the Add Expression dialog box. Your expression is inserted into the Expression text area.

1. Do one of the following:
   - Navigate to Security > Application Firewall, and in the details pane, click Application Firewall policy manager.
   - Navigate to Security > Application Firewall > Policies > Firewall Policies, and in the details pane, click Policy Manager.
2. In the Application Firewall Policy Manager dialog, choose the bind point to which you want to bind the policy from the drop-down list. The choices are:
   - **Override Global.** Policies that are bound to this bind point process all traffic from all interfaces on the NetScaler appliance, and are applied before any other policies.
   - **LB Virtual Server.** Policies that are bound to a load balancing virtual server are applied only to traffic that is processed by that load balancing virtual server, and are applied before any Default Global policies. After selecting LB Virtual Server, you must also select the specific load balancing virtual server to which you want to bind this policy.
   - **CS Virtual Server.** Policies that are bound to a content switching virtual server are applied only to traffic that is processed by that content switching virtual server, and are applied before any Default Global policies. After selecting CS Virtual Server, you must also select the specific content switching virtual server to which you want to bind this policy.
   - **Default Global.** Policies that are bound to this bind point process all traffic from all interfaces on the NetScaler appliance.
   - **Policy Label.** Policies that are bound to a policy label process traffic that the policy label routes to them. The policy label controls the order in which policies are applied to this traffic.
   - **None.** Do not bind the policy to any bind point.
3. Select Insert Policy to insert a new row and display a drop-down list with all available, unbound application firewall policies.
4. Select the policy you want to bind, or select New Policy to create a new policy. The policy that you selected or created is inserted into the list of globally bound application firewall policies.
5. Make any additional adjustments to the binding.
   - To modify the policy priority, click the field to enable it, and then type a new priority. You can also select Regenerate Priorities to renumber the priorities evenly.
   - To modify the policy expression, double click that field to open the Configure Application Firewall Policy dialog box, where you can edit the policy expression.
   - To set the Goto Expression, double click field in the Goto Expression column heading to display the drop-down list, where you can choose an expression.
   - To set the Invoke option, double click field in the Invoke column heading to display the drop-down list, where you can choose an expression
6. Repeat steps 3 through 6 to add any additional application firewall policies you want to globally bind.
7. Click OK. A message appears in the status bar, stating that the policy has been successfully bound.

Manual Configuration By Using the Command Line Interface

Dec 17, 2013

You can configure many application firewall features from the NetScaler command line. There are important exceptions, however. You cannot enable signatures from the command line. There are over 1,000 default signatures in seven categories; the task is simply too complex for the command line interface. You can configure the check actions and parameters for security checks from the command line, but cannot enter manual relaxations. While you can configure the adaptive learning feature and enable learning from the command line, you cannot review learned relaxations or learned rules and approve or skip them. The command line interface is intended for advanced users who are thoroughly familiar with the NetScaler appliance and the application firewall feature.

To manually configure the application firewall by using the NetScaler command line, use a telnet or secure shell client of your choice to log on to the NetScaler command line.

At the command prompt, type the following commands:

- add appfw profile <name> [-defaults (**basic** | **advanced**)]
- set appfw profile <name> -type (**HTML** | **XML** | **HTML XML**)
- save ns config

Example

The following example adds a profile named **pr-basic**, with basic defaults, and assigns a profile type of **HTML**. This is the appropriate initial configuration for a profile to protect an HTML Web site.

```
add appfw profile pr-basic -defaults basic
set appfw profile pr-basic -type HTML
save ns config
```

At the command prompt, type the following commands:

- set appfw profile <name> <arg1> [<arg2> ...] where <arg1> represents a parameter and <arg2> represents either another parameter or the value to assign to the parameter represented by <arg1>. For descriptions of the parameters to use when configuring specific security checks, see Advanced Protections and its subtopics. For descriptions of the other parameters, see "Parameters for Creating a Profile."
- save ns config

Example

The following example shows how to configure an HTML profile created with basic defaults to begin protecting a simple HTML-based Web site. This example turns on logging and maintenance of statistics for most security checks, but enables blocking only for those checks that have extremely low false positive rates and require no special configuration. It also turns on transformation of unsafe HTML and unsafe SQL, which prevents attacks but does not block requests to your Web sites. With logging and statistics enabled, you can later review the logs to determine whether to enable blocking for a

specific security check.

```
set appfw profile -startURLAction log stats
set appfw profile -denyURLAction block log stats
set appfw profile -cookieConsistencyAction log stats
set appfw profile -crossSiteScriptingAction log stats
set appfw profile -crossSiteScriptingTransformUnsafeHTML ON
set appfw profile -fieldConsistencyAction log stats
set appfw profile -SQLInjectionAction log stats
set appfw profile -SQLInjectionTransformSpecialChars ON
set appfw profile -SQLInjectionOnlyCheckFieldsWithSQLChars ON
set appfw profile -SQLInjectionParseComments checkall
set appfw profile -fieldFormatAction log stats
set appfw profile -bufferOverflowAction block log stats
set appfw profile -CSRFtagAction log stats
save ns config
```

At the command prompt, type the following commands:

- add appfw policy <name> <rule> <profile>
- save ns config

Example

The following example adds a policy named **pl-blog**, with a rule that intercepts all traffic to or from the host **blog.example.com**, and associates that policy with the profile **pr-blog**. This is an appropriate policy to protect a blog hosted on a specific hostname.

```
add appfw policy pl-blog "HTTP.REQ.HOSTNAME.DOMAIN.EQ("blog.example.com")" pr-blog
```

At the command prompt, type the following commands:

- bind appfw global <policyName> <priority>
- save ns config

Example

The following example binds the policy named **pl-blog** and assigns it a priority of **10**.

```
bind appfw global pl-blog 10
save ns config
```

Signatures

May 02, 2013

The application firewall signatures function provides specific, configurable rules that protect your web sites against known attacks. A signature represents a pattern that is a component of a known attack on an operating system, a web server, a web site, an XML-based web service, or any other server that is connected to a web site or web service. A signature can be used to check either requests or responses. A signature can consist of a literal string or a PCRE-compliant regular expression.

To specify how the application firewall is to use signatures, you configure a signatures object, which specifies the signatures to apply to your traffic and the actions to be taken when the signatures match the traffic. A signatures object also contains the SQL injection and cross-site scripting patterns, and may also contain XPath injection patterns. These patterns are not actually signatures but are used by some of the advanced protection checks. The SQL Injection and Cross-Site Scripting patterns contain the SQL special symbols and keywords, the cross-site scripting allowed tags and attributes, and the denied patterns for the HTML and XML SQL Injection and Cross-Site Scripting checks. The XPath injection patterns contain the XPath (XML Path Language) denied patterns.

Note: If you use the wizard to configure signatures, the signatures object is created automatically.
The application firewall examines requests to your protected web sites and web services to determine whether a request matches a signature. Matching requests are handled as you specify when configuring the Signatures actions. By default, matching requests are logged so that you can examine them later. If you enabled blocking, the application firewall displays an error page or error object. If you enabled statistics, the application firewall also includes the request in the statistics that it maintains about requests that match an application firewall signature or security check.

If you want to configure signatures manually, you must create a signatures object from a template or import a signatures object file. There are two default templates that you can use: the *Default Signatures template and the *Xpath Injection template. The *Default Signatures template contains over 1,000 signatures, in addition to the complete list of SQL injection and cross-site scripting allowed and denied patterns. The *Xpath Injection template contains all of those, and in addition contains 57 Xpath keywords and special strings.

In addition to using its native signatures format, the application firewall can create a signatures object by using a built-in template for any supported external signatures format, or by importing an external signatures file in a supported format. The supported formats are as follows:

- **Cenzic**—Signatures files, produced by Cenzic products, that use Cenzic Hailstorm technology.
- **IBM AppScan**—Signatures files produced by IBM AppScan Enterprise and IBM AppScan Standard.
- **Qualys**—Qualys WAS signatures files produced by QualysGuard products. Only Qualys WAS 1.0 files are supported for importing as signatures. WAS 2.0 is not supported.

  Note: Qualys classifies a single SQL special character in a URL as a security threat, even when no SQL keywords are present. The SQL injection check does not consider the presence of a single SQL special character a threat unless an SQL keyword is present. For that reason, a Qualys scanner continues to report such requests as containing SQL injection vulnerabilities, but the application firewall does not detect or block these requests because they pose no actual threat to your protected web sites and web services.
- **Trend Micro**—Signatures files produced by the Trend Micro Vulnerability Scanner (TMVS).

- **Whitehat**—WASC 1.0, WASC 2.0, and best practices signatures produced by Whitehat Sentinel products.

  WASC signatures include information about many vulnerabilities. The application firewall generates blocking signatures from all WASC vulnerabilities. However, only certain vulnerabilities are appropriate for the web application firewall environment. For a list of appropriate Whitehat signatures, see Whitehat WASC Signature Types for WAF Use.

Once you have created a signatures object, you can configure all parts of it, including the signatures rules, the XML SQL Injection and Cross-Site Scripting rules, and the Xpath injection rules. You can manually create and modify your own custom signatures in the signatures editor. You can also add new SQL injection, cross-site scripting, and XPath injection patterns, modify existing patterns, and remove patterns.

Regardless of whether you use the wizard for initial configuration or configure your signatures object manually, you should regularly apply the Citrix updates to keep your signatures current. Citrix regularly updates the default application firewall signatures. You can apply those updates manually, or you can enable automatic signature updates so that the application firewall can update the signatures from the Cloud-based application firewall updates service. You can obtain the correct URL for either type of updates from your Citrix service representative or reseller.

Manually Configuring the Signatures Feature

Sep 03, 2013

To use signatures to protect your web sites, you must review the rules, and enable and configure the ones that you want to apply. The rules are disabled by default. Citrix recommends that you enable all rules that are applicable to the type of content that your web site uses.

To manually configure the signatures feature, use a browser to connect to the configuration utility. Then, create a signatures object from a built-in template, an existing signatures object, or by importing a file. Next, configure the new signatures object.

Note: The following procedures do not address adding user-defined signatures to a signatures object. To create your own signatures, see "The Signatures Editor."

Adding or Removing a Signatures Object

Oct 06, 2014

You can add a new signatures object to the application firewall by:

- Copying a built-in template.
- Copying an existing signatures object.
- Importing a signatures object from an external file.

You must use the configuration utility to copy a template or existing signatures object. You can use either the configuration utility or the command line to import a signatures object. You can also use either the configuration utility or the command line to remove a signatures object.

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to use as a template.
   Your choices are:
   - *** Default Signatures.** Contains the signatures rules, the SQL injection rules, and the cross-site scripting rules.
   - *** XPath Injection.** Contains the XPath injection patterns.
   - **Any existing signatures object.**

   Attention: If you do not choose a signatures type to use as a template, the application firewall will prompt you to create signatures from scratch.
3. Click Add.
4. In the Add Signatures Object dialog box, type a name for your new signatures object, and then click OK. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), and underscore (_) symbols.
5. Click Close.

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, click Add.
3. In the Add Signatures Object dialog box, select the format of the signatures you want to import.
   - To import a NetScaler format signatures file, select the Native Format tab.
   - To import an external signatures format file, select the External Format tab.
4. Choose the file that you want to use to create your new signatures object.
   - To import a native NetScaler format signatures file, in the Import section select either Import from Local File or Import from URL, then type or browse to the path or URL to the file.
   - To import a Cenzic, IBM AppScan, Qualys, or Whitehat format file, in the XSLT section select Use Built-in XSLT File, Use Local File, or Reference from URL. Next, if you chose Use Built-in XSLT File, select the appropriate file format from the drop-down list. If you chose Use Local File or Reference from URL, then type or browse to the path or URL to the file.
5. Click Add, and then click Close.

At the command prompt, type the following commands:

- import appfw signatures <src> <name> [-xslt <string>] [-comment <string>] [-overwrite] [-merge] [-sha1 <string>]
- save ns config

Example #1

The following example creates a new signatures object from a file named signatures.xml and assigns it the name MySignatures.

```
import appfw signatures signatures.xml MySignatures
save ns config
```

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to remove.
3. Click Remove.

At the command prompt, type the following commands:

- rm appfw signatures <name>
- save ns config

Configuring or Modifying a Signatures Object

Oct 06, 2014

You configure a signatures object after creating it, or modify an existing signatures object, to enable or disable signature categories or specific signatures, and configure how the application firewall responds when a signature matches a connection.

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to configure, and then click Open.
3. In the Modify Signatures Object dialog box, set the Display Filter Criteria options at the left to display the filter items that you want to configure.
   As you modify these options, the results that you requested are displayed in the Filtered Results window at the right.

- To display only selected categories of signatures, check or clear the appropriate signature-category check boxes. The signature categories are:

Name	Type of Attack that this Signature Protects Against
cgi	CGI scripts. Includes Perl and UNIX shell scripts.
client	Browsers and other clients.
coldfusion	Web sites that use the Adobe Systems ColdFusion application server.
frontpage	Web sites that use Microsoft's FrontPage server.
iis	Web sites that use the Microsoft Internet Information Server (IIS).
misc	Miscellaneous attacks.
php	Web sites that use PHP
web-activex	Web sites that contain ActiveX controls.
web-struts	Web sites that contain Apache struts, which are java-ee based applets.

- To display only signatures that have specific check actions enabled, select the ON check box for each of those actions, clear the ON check boxes for the other actions, and clear all of the OFF check boxes. To display only signatures that have a specific check action disabled, select their respective OFF check boxes and clear all of the ON check boxes. To display signatures regardless of whether they have a check action enabled or disabled, select or clear both the ON and the OFF check boxes for that action. The check actions are:

Criterion	Description
Enabled	The signature is enabled. The application firewall checks only for signatures that are enabled when it processes traffic.
Block	Connections that match this signature are blocked.
Log	A log entry is produced for any connection that matches this signature.
Stats	The application firewall includes any connection that matches this signature in the statistics that it generates for that check.

- To display only signatures that contain a specific string, type the string into the text box under the filter criteria, and then click Search.
- To reset all display filter criteria to the default settings and display all signatures, click Show All.

4. For information about a specific signature, select the signature, and then click the blue double arrow in the More field. The Signature Rule Vulnerability Detail message box appears. It contains information about the purpose of the signature and provides links to external web-based information about the vulnerability or vulnerabilities that this signature addresses. To access an external link, click the blue double arrow to the left of the description of that link.
5. Configure the settings for a signature by selecting the appropriate check boxes.
6. If you want to add a local signature rule to the signatures object, or modify an existing local signature rule, see "The Signatures Editor."
7. If you have no need for SQL injection, cross-site scripting, or Xpath injection patterns, click OK, and then click Close. Otherwise, in the lower left-hand corner of the details pane, click Manage SQL/XSS Patterns.
8. In the Manage SQL/XSS Patterns dialog box, Filtered Results window, navigate to the pattern category and pattern that you want to configure. For information about the SQL injection patterns, see "HTML SQL Injection Check." For information about the cross-site scripting patterns, see "HTML Cross-Site Scripting Check."
9. To add a new pattern:
    1. Select the branch to which you want to add the new pattern.
    2. Click the Add button directly below the lower section of the Filtered Results window.
    3. In the Create Signature Item dialog box, fill in the Element text box with the pattern that you want to add. If you are adding a transformation pattern to the transform rules branch, under Elements, fill in the From text box with the pattern that you want to change and the To text box with the pattern to which you want to change the previous pattern.
    4. Click OK.
10. To modify an existing pattern:
    1. In the Filtered Results window, select the branch that contains the pattern that you want to modify.
    2. In the detail window beneath the Filtered Results window, select the pattern that you want to modify.
    3. Click Modify.
    4. In the Modify Signature Item dialog box, Element text box, modify the pattern. If you are modifying a transformation pattern, you can modify either or both patterns under Elements, in the From and the To text boxes.
    5. Click OK.
11. To remove a pattern, select the pattern that you want to remove, then click the Remove button below the details pane beneath the Filtered Results window. When prompted, confirm your choice by clicking Close.
12. To add the patterns category to the XSS branch:

1. Select the branch to which you want to add the patterns category.
2. Click the Add button directly below the Filtered Results window.
   Note: Currently you can add only one category, named patterns, to the XSS branch, so after you click Add, you must accept the default choice, which is patterns.
3. Click OK.

13. To remove a branch, select that branch, and then click the Remove button directly below the Filtered Results window. When prompted, confirm your choice by clicking OK.
    Note: If you remove a default branch, you remove all of the patterns in that branch. Doing so can disable the security checks that use that information.
14. When you are finished modifying the SQL injection, cross-site scripting, and XPath injection patterns, click OK, and then click Close to return to the Modify Signatures Object dialog box.
15. Click OK at any point to save your changes, and when you are finished configuring the signatures object, click Close.

Updating a Signatures Object

Jul 21, 2015

You should update your signatures objects frequently to ensure that your application firewall is providing protection against current threats. You should regularly update both the default application firewall signatures and any signatures that you import from a supported vulnerability scanning tool.

Citrix regularly updates the default signatures for the application firewall. You can update the default signatures manually or automatically. In either case, ask your Citrix representative or Citrix reseller for the URL to access the updates. You can enable automatic updates of the Citrix native format signatures in the "Engine Settings" and "Signature Auto Update Settings" dialog boxes.

Most makers of vulnerability scanning tools regularly update the tools. Most web sites also change frequently. You should update your tool and rescan your web sites regularly, exporting the resulting signatures to a file and importing them into your application firewall configuration.

Note: When you update the application firewall signatures from the NetScaler command line, you must first update the default signatures, and then issue additional update commands to update each custom signatures file that is based on the default signatures. If you do not update the default signatures first, a version mismatch error prevents updating of the custom signatures files.

At the command prompt, type the following commands:

- update appfw signatures <name> [-mergedefault]
- save ns config

Example

The following example updates the signatures object named MySignatures from the default signatures object, merging new signatures in the default signatures object with the existing signatures. This command does not overwrite any user-created signatures or signatures imported from another source, such as an approved vulnerability scanning tool.

```
update appfw signatures MySignatures -mergedefault
save ns config
```

Updated: 2014-10-06

Citrix regularly updates the signatures for the Application Firewall. You should regularly update the signatures on your Application Firewall to ensure that your Application Firewall is using the most current list. Ask your Citrix representative or Citrix reseller for the URL to access the updates.

To update a signatures object from a Citrix format file by using the command line

At the command prompt, type the following commands:

- update appfw signatures <name> [-mergeDefault]
- save ns config

To update a signatures object from a Citrix format file by using the configuration utility

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to update.
3. In the Action drop-down list, select Merge.
4. In the Update Signatures Object dialog box, choose one of the following options.
   - **Import from URL**—Choose this option if you download signature updates from a web URL.
   - **Import from Local File**—Choose this option if you import signature updates from a file on your local hard drive, network hard drive, or other storage device.
5. In the text area, type the URL, or type or browse to the local file.
6. Click Update. The update file is imported, and the Update Signatures dialog box changes to a format nearly identical to that of the Modify Signatures Object dialog box. The Update Signatures Object dialog box displays all branches with new or modified signature rules, SQL injection or cross-site scripting patterns, and XPath injection patterns if there are any.
7. Review and configure the new and modified signatures.
8. When you are finished, click OK, and then click Close.

Updated: 2014-01-17

Note: Before you update a signatures object from a file, you must create the file by exporting signatures from the vulnerability scanning tool.

To import and update signatures from a vulnerability scanning tool

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to update, and then click Merge.
3. In the Update Signatures Object dialog box, on the External Format tab, Import section, choose one of the following options.
   - **Import from URL**—Choose this option if you download signature updates from a Web URL.
   - **Import from Local File**—Choose this option if you import signature updates from a file on your local or a network hard drive or other storage device.
4. In the text area, type the URL, or browse or type the path to the local file.
5. In the XSLT section, choose one of the following options.
   - **Use Built-in XSLT File**—Choose this option if you want to use a built-in XSLT files.
   - **Use Local File**—Choose this option to use an XSLT file on your local computer.
   - **Reference from URL**—Choose this option to import an XSLT file from a web URL.
6. If you chose Use Built-in XSLT File, in the Built-In XSLT drop-down list choose the built-in XSLT file that you want to use.
   - To use the Cenzic XSLT file, select Cenzic.
   - To use the IBM AppScan Standard XSLT file, select IBM AppScan Standard.
   - To use the IBM AppScan Enterprise XSLT file, select IBM AppScan Enterprise.
   - To use the Qualys XSLT file, select Qualys.
   - To use the Trend Microsystems XSLT file, select Trend Micro.
   - To use the Whitehat XSLT file, select Whitehat.
7. Click Update. The update file is imported, and the Update Signatures dialog box changes to a format nearly identical to that of the Modify Signatures Object dialog box, which is described in "Configuring or Modifying a Signatures Object."

The Update Signatures Object dialog box displays all branches with new or modified signature rules, SQL injection or cross-site scripting patterns, and XPath injection patterns if there are any.

8. Review and configure the new and modified signatures.
9. When you are finished, click OK, and then click Close.

Exporting a Signatures Object to a File

Oct 06, 2014

You export a signatures object to a file so that you can import it to another NetScaler ADC.

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to configure.
3. In the Actions drop-down list, select Export.
4. In the Export Signatures Object dialog box, Local File text box, type the path and name of the file to which you want to export the signatures object, or use the Browse dialog to designate a path and name.
5. Click OK.

The Signatures Editor

Oct 14, 2014

You can use the signatures editor, which is available in the configuration utility, to add a new user-defined (local) signature rule to an existing signatures object, or to modify a previously configured local signature rule. Except that it is defined by the user (you), a local signature rule has the same attributes as a default signature rule from Citrix, and it functions in the same way. You enable or disable it, and configure the signature actions for it, just as you do for a default signature.

Add a local rule if you need to protect your web sites and services from a known attack that the existing signatures do not match. For example, you might discover a new type of attack and determine its characteristics by examining the logs on your web server, or you might obtain third-party information about a new type of attack.

At the heart of a signature rule are the rule *patterns*, which collectively describe the characteristics of the attack that the rule is designed to match. Each pattern can consist of a simple string, a PCRE-format regular expression, or the built-in SQL injection or cross-site scripting patterns.

You might want to modify a signature rule by adding a new pattern or modifying an existing pattern to match an attack. For example, you might find out about changes to an attack, or you might determine a better pattern by examining the logs on your web server, or from third-party information.

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select the signatures object that you want to edit, and then click Open.
3. In the Modify Signatures Object dialog box, in the middle of the screen beneath the Filtered Results window, do one of the following:.
   - To add a new local signature rule, click Add.
   - To modify an existing local signature rule, select that rule, and then click Open.
4. In the Add Local Signature Rule or the Modify Local Signature Rule dialog box, configure the actions for a signature by selecting the appropriate check boxes.
   - **Enabled.** Enables the new signature rule. If you do not select this, this new signature rule is added to your configuration, but is inactive.
   - **Block.** Blocks connections that violate this signature rule.
   - **Log.** Logs violations of this signature rule to the NetScaler log.
   - **Stat.** Includes violations of this signature rule in the statistics.
   - **Remove.** Strips information that matches the signature rule from the response. (Applies only to response rules.)
   - **X-Out.** Masks information that matches the signature rule with the letter X. (Applies only to response rules.)
   - **Allow Duplicates.** Allows duplicates of this signature rule in this signatures object.
5. Choose a category for the new signature rule from the Category drop-down list.
   You can also create a new category by clicking the icon to the right of the list and using the Add Signature Rule Category dialog box to add a new category to the list, The rule you are modifying is automatically added to the new category. For instructions, see "To add a signature rule category."
6. In the **LogString** text box, type a brief description of the signature rule to be used in the logs.
7. In the **Comment** text box, type a comment. (Optional)
8. Click More..., and modify the advanced options.
   1. To strip HTML comments before applying this signature rule, in the Strip Comments drop-down list choose All or

Exclude Script Tag.

2. To set CSRF Referer Header checking, in the CSRF Referer Header checking radio button array, select either the If Present or Always radio button.
3. To manually modify the Rule ID assigned to this local signature rule, modify the number in the Rule ID text box. The ID must be a positive integer between 1000000 and 1999999 that has not already been assigned to a local signature rule.
4. To assign a version number to the new signature rule, modify the number in the Version Number text box.
5. To assign a Source ID, modify the string in the Source ID text box.
6. To specify the source, choose Local or Snort from the Source drop-down list, or click the Add icon to the right of the list and add a new source.
7. To assign a harm score to violations of this local signature rule, type a number between 1 and 10 in the Harm Score text box.
8. To assign a severity rating to this local signature rule, in the Severity drop-down list choose High, Medium, or Low, or click the Add icon to the right of the list and add a new severity rating.
9. To assign a violation type to this local signature rule, in the Type drop-down list choose Vulnerable or Warning, or click the Add icon to the right of the list and add a new violation type.

9. In the **Patterns** list, add or edit a pattern.
   - To add a pattern, click Add. In the Create New Signature Rule Pattern dialog box, add one or more patterns for your signature rule, and then click OK.
   - To edit a pattern, select the pattern, and then click Open. In the Edit Signature Rule Pattern dialog box, modify the pattern, and then click OK.

   For more information about adding or editing patterns, see "Signature Rule Patterns."
10. Click OK.

To add a signature rule category

Sep 03, 2013

Putting signature rules into a category enables you to configure the actions for a group of signatures instead of for each individual signature. You might want to do so for the following reasons:

- **Ease of selection.** For example, assume that all of signature rules in a particular group protect against attacks on a specific type of web server software or technology. If your protected web sites use that software or technology, you want to enable them all. If they do not, you do not want to enable any of them.
- **Ease of initial configuration.** It is easiest to set defaults for a group of signatures as a category, instead of one-by-one. You can then make any changes to individual signatures as needed.
- **Ease of ongoing configuration.** It is easier to configure signatures if you can display only those that meet specific criteria, such as belonging to a specific category.

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select that signatures object that you want to configure, and then click Open.
3. In the Modify Signatures Object dialog box, in the middle of the screen, beneath the Filtered Results window, click Add.
4. In the Add Local Signature Rule dialog box, click the icon to the right of the Category drop-down list.
5. In the Add Signature Rule Category dialog box, New Category text box, type a name for your new signature category. The name can consist of from one to 64 characters.
6. Click **OK**.

Signature Rule Patterns

Aug 20, 2014

You can add a new pattern to a signature rule or modify an existing pattern of a signature rule to specify a string or expression that characterizes an aspect of the attack that the signature matches. To determine which patterns an attack exhibits, you can examine the logs on your web server, use a tool to observe connection data in real time, or obtain the string or expression from a third-party report about the attack.

Caution: Any new pattern that you add to a signature rule is in an **AND** relationship with the existing patterns. Do not add a new pattern to an existing signature rule if you do not want a potential attack to have to match all of the patterns in order to match the signature.

Each pattern can consist of a simple string, a PCRE-format regular expression, or the built-in SQL injection or cross-site scripting pattern. Before you attempt to add a pattern that is based on a regular expression, you should make sure that you understand PCRE-format regular expressions. PCRE expressions are complex and powerful; if you do not understand how they work, you can unintentionally create a pattern that matches something that you did not want (a *false positive*) or that fails to match something that you did want (a *false negative*).

If you are not already familiar with PCRE-format regular expressions, you can use the following resources to learn the basics, or for help with some specific issue:

- *— "Mastering Regular Expressions"*
  , Third Edition. Copyright (c) 2006 by Jeffrey Friedl. O'Reilly Media, ISBN: 9780596528126
- *— "Regular Expressions Cookbook"*
  . Copyright (c) 2009 by Jan Goyvaerts and Steven Levithan. O'Reilly Media, ISBN: 9780596520687
- **PCRE Man page/Specification** (text/official): "http://www.pcre.org/pcre.txt"
- **PCRE Man Page/Specification** (html/gammon.edu.au): "http://www.gammon.com.au/pcre/index.html"
- **Wikipedia PCRE entry**: "http://en.wikipedia.org/wiki/PCRE"
- **PCRE Mailing List** (run by exim.org): "http://lists.exim.org/mailman/listinfo/pcre-dev"

If you need to encode non-ASCII characters in a PCRE-format regular expression, the NetScaler platform supports encoding of hexadecimal UTF-8 codes. For more information, see "PCRE Character Encoding Format."

1. Navigate to Security > Application Firewall > Signatures.
2. In the details pane, select that signatures object that you want to configure, and then click **Open**.
3. In the Modify Signatures Object dialog box, in the middle of the screen beneath the Filtered Results window, either click Add to create a signature rule, or select an existing signature rule and click Open.
   Note: You can modify only signature rules that you added. You cannot modify the default signature rules.
   Depending on your action, either the Add Local Signature Rule or the Modify Local Signature Rule dialog box appears. Both dialog boxes have the same contents.
4. Under the Patterns window in the dialog box, either click Add to add a new pattern, or select an existing pattern from the list beneath the Add button and click Open. Depending on your action, either the Create New Signature Rule Pattern or the Edit Signature Rule Pattern dialog box appears. Both dialog boxes have the same contents.
5. In the Pattern Type drop-down list, choose the type of connection that the pattern is intended to match.
   - If the pattern is intended to match request elements or features, such as injected SQL code, attacks on web forms, cross-site scripts, or inappropriate URLs, choose **Request**.
   - If the pattern is intended to match response elements or features, such as credit card numbers or safe objects,

choose **Response**.

6. In the Location area, define the elements to examine with this pattern.
   The Location area describes what elements of the HTTP request or response to examine for this pattern. The choices that appear in the Location area depend upon the chosen pattern type. If you chose **Request** as the pattern type, items relevant to HTTP requests appear; if you chose **Response**, items relevant to HTTP responses appear.

   In addition, as you choose a value from the Area drop-down list, the remaining parts of the Location area change interactively. Following are all configuration items that might appear in this section.

   Area

   Drop-down list of elements that describe a particular portion of the HTTP connection. The choices are as follows:
   - **HTTP_ANY**. All parts of the HTTP connection.
   - **HTTP_COOKIE**. All cookies in the HTTP request headers after any cookie transformations are performed.
     Note: Does not search HTTP response "Set-Cookie:" headers.
   - **HTTP_FORM_FIELD**. Form fields and their contents, after URL decoding, percent decoding, and removal of excess whitespace. You can use the **<Location>** tag to further restrict the list of form field names to be searched.
   - **HTTP_HEADER**. The value portions of the HTTP header after any cross-site scripting or URL decoding transformations.
   - **HTTP_METHOD**. The HTTP request method.
   - **HTTP_ORIGIN_URL**. The origin URL of a web form.
   - **HTTP_POST_BODY**. The HTTP post body and the web form data that it contains.
   - **HTTP_RAW_COOKIE**. All HTTP request cookie, including the "Cookie:" name portion.
     Note: Does not search HTTP response "Set-Cookie:" headers.
   - **HTTP_RAW_HEADER**. The entire HTTP header, with individual headers separated by linefeed characters (\n) or carriage return/line-feed strings (\r\n).
   - **HTTP_RAW_RESP_HEADER**. The entire response header, including the name and value parts of the response header after URL transformation has been done, and the complete response status. As with HTTP_RAW_HEADER, individual headers are separated by linefeed characters (\n) or carriage return/line-feed strings (\r\n).
   - **HTTP_RAW_SET_COOKIE**. The entire Set-Cookie header after any URL transformations have been performed.
     Note: URL transformation can change both the domain and path parts of the Set-Cookie header.
   - **HTTP_RAW_URL**. The entire request URL before any URL transformations are performed, including any query or fragment parts.
   - **HTTP_RESP_HEADER**. The value part of the complete response headers after any URL transformations have been performed.
   - **HTTP_RESP_BODY**. The HTTP response body.
   - **HTTP_SET_COOKIE**. All "Set-Cookie" headers in the HTTP response headers.
   - **HTTP_STATUS_CODE**. The HTTP status code.
   - **HTTP_STATUS_MESSAGE**. The HTTP status message.
   - **HTTP_URL**. The value portion of the URL in the HTTP headers, excluding any query or fragment ports, after conversion to the UTF-* character set, URL decoding, stripping of whitespace, and conversion of relative URLs to absolute. Does not include HTML entity decoding.

   URL

   Examines any URLs found in the elements specified by the Area setting. Select one of the following settingss.
   - **Any.** Checks all URLs.
   - Literal. Checks URLs that contain a literal string. After you select Literal, a text box is displayed. Type the literal string that you want in the text box.
   - PCRE. Checks URLs that match a PCRE-format regular expression. After you select this choice, the regular expression

window is displayed. Type the regular expression in the window. You can use the **Regex Tokens** to insert common regular expression elements at the cursor, or you can click Regex Editor to display the Regular Expression Editor dialog box, which provides more assistance in constructing the regular expression that you want.

- Expression. Checks URLs that match a NetScaler default expression.

Field Name

Examines any form field names found in the elements specified by the Area selection.

- **Any.** Checks all URLs.
- Literal. Checks URLs that contain a literal string. After you select Literal, a text box is displayed. Type the literal string that you want in the text box.
- PCRE. Checks URLs that match a PCRE-format regular expression. After you select this choice, the regular expression window is displayed. Type the regular expression in the window. You can use the **Regex Tokens** to insert common regular expression elements at the cursor, or you can click Regex Editor to display the Regular Expression Editor dialog box, which provides more assistance in constructing the regular expression that you want.
- Expression. Checks URLs that match a NetScaler default expression.

7. In the Pattern area, define the pattern. A pattern is a literal string or PCRE-format regular expression that defines the pattern that you want to match. The Pattern area contains the following elements:

   Match

   A drop-down list of search methods that you can use for the signature. This list differs depending on whether the pattern type is Request or Response.

   Request Match Types

   - **Literal**. A literal string.
   - **PCRE**. A PCRE-format regular expression.
     NOTE: When you choose PCRE, the regular expression tools beneath the Pattern window are enabled. These tools are not useful for most other types of patterns.
   - **Injection**. Directs the application firewall to look for injected SQL in the specified location. The Pattern window disappears, because the application firewall already has the patterns for SQL injection.
   - **CrossSiteScripting**. Directs the application firewall to look for cross-site scripts in the specified location. The Pattern window disappears, because the application firewall already has the patterns for cross-site scripts.
   - **Expression**. An expression in the NetScaler default expressions language. This is the same expressions language that is used to create application firewall policies and other policies on the NetScaler appliance. Although the NetScaler expressions language was originally developed for policy rules, it is a highly flexible general purpose language that can also be used to define a signature pattern.
     When you choose Expression, the NetScaler Expression Editor appears beneath Pattern window. For more information about the Expression Editor and instructions on how to use it, see "To add a firewall rule (expression) by using the Add Expression dialog box." For more information about NetScaler expressions, see "Policies and Expressions."

   Response Match Types

   - **Literal**. A literal string.
   - **PCRE**. A PCRE-format regular expression.
     NOTE: When you choose PCRE, the regular expression tools beneath the Pattern window are enabled. These tools are not useful for most other types of patterns.
   - **Credit Card**. A built-in pattern to match one of the six supported types of credit card number.

Note: The Expression match type is not available for Response-side signatures.

Pattern Window (unlabeled)

In this window, type the pattern that you want to match, and fill in any additional data.

- **Literal.** Type the string you want to search for in the text area.
- **PCRE.** Type the regular expression in the text area. Use the **Regex Editor** for more assistance in constructing the regular expression that you want, or the Regex Tokens to insert common regular expression elements at the cursor. To enable UTF-8 characters, click UTF-8.
- **Expression.** Type the NetScaler advanced expression in the text area. Use Prefix to choose the first term in your expression, or Operator to insert common operators at the cursor. Click **Add** to open the Add Expression dialog box for more assistance in constructing the regular expression that you want. Click Evaluate to open the Advanced Expression Evaluator to help determine what effect your expression has.
- **Offset.** The number of characters to skip over before starting to match on this pattern. You use this field to start examining a string at some point other than the first character.
- **Depth.** How many characters from the starting point to examine for matches. You use this field to limit searches of a large string to a specific number of characters.
- **Min-Length.** The string to be searched must be at least the specified number of bytes in length. Shorter strings are not matched.
- **Max-Length.** The string to be searched must be no longer than the specified number of bytes in length. Longer strings are not matched.
- **Search method.** A check box labeled fastmatch. You can enable fastmatch only for a literal pattern, to improve performance.

8. Click OK.
9. Repeat the previous four steps to add or modify additional patterns.
10. When finished adding or modifying patterns, click OK to save your changes and return to the Signatures pane.
    Caution: Until you click **OK** in the **Add Local Signature Rule** or **Modify Local Signature Rule** dialog box, your changes are not saved. Do not close either of these dialog boxes without clicking **OK** unless you want to discard your changes.

Overview of Security checks

Sep 03, 2013

The application firewall advanced protections (security checks) are a set of filters designed to catch complex or unknown attacks on your protected web sites and web services. The security checks use heuristics, positive security, and other techniques to detect attacks that may not be detected by signatures alone. You configure the security checks by creating and configuring an application firewall profile, which is a collection of user-defined settings that tell the application firewall which security checks to use and how to handle a request or response that fails a security check. A profile is associated with a signatures object and with a policy to create a security configuration.

The application firewall provides twenty security checks, which differ widely in the types of attacks that they target and how complex they are to configure. The security checks are organized into the following categories:

- **Common security checks.** Checks that apply to any aspect of web security that either does not involve content or is equally applicable to all types of content.
- **HTML security checks.** Checks that examine HTML requests and responses. These checks apply to HTML-based web sites and to the HTML portions of Web 2.0 sites, which contain mixed HTML and XML content.
- **XML security checks.** Checks that examine XML requests and responses. These checks apply to XML-based web services and to the XML portions of Web 2.0 sites.

The security checks protect against a wide range of types of attack, including attacks on operation system and web server software vulnerabilities, SQL database vulnerabilities, errors in the design and coding of web sites and web services, and failures to secure sites that host or can access sensitive information.

All security checks have a set of configuration options, the check actions, which control how the application firewall handles a connection that matches a check. Three check actions are available for all security checks. They are:

- **Block.** Block connections that match the signature. Disabled by default.
- **Log.** Log connections that match the signature, for later analysis. Enabled by default.
- **Stats.** Maintain statistics, for each signature, that show how many connections it matched and provide certain other information about the types of connections that were blocked. Disabled by default.

A fourth check action, **Learn**, is available for more than half of the check actions. It observes traffic to a protected Web site or web service and uses connections that repeatedly violate the security check to generate recommended exceptions (relaxations) to the check, or new rules for the check. In addition to the check actions, certain security checks have parameters that control the rules that the check uses to determine which connections violate that check, or that configure the application firewall's response to connections that violate the check. These parameters are different for each check, and they are described in the documentation for each check.

To configure security checks, you can use the application firewall wizard, as described in "The Application Firewall Wizard," or you can configure the security checks manually, as described in "Manual Configuration By Using the Configuration Utility." Some tasks, such as manually entering relaxations or rules or reviewing learned data, can be done only by using the configuration utility, not the command line. Using the wizard is usually best configuration method, but in some cases manual configuration might be easier if you are thoroughly familiar with it and simply want to adjust the configuration for a single security check.

Regardless of which method you use to configure the security checks, each security check requires that certain tasks be performed. Many checks require that you specify exceptions (relaxations) to prevent blocking of legitimate traffic before

you enable blocking for that security check. You can do this manually, by observing the log entries after a certain amount of traffic has been filtered and then creating the necessary exceptions. However, it is usually much easier to enable the learning feature and let it observe the traffic and recommend the necessary exceptions.

Top-Level Protections

Oct 01, 2013

Four of the application firewall protections are especially effective against common types of Web attacks, and are therefore more commonly used than any of the others. They are:

- **HTML Cross-Site Scripting.** Examines requests and responses for scripts that attempt to access or modify content on a different Web site than the one on which the script is located. When this check finds such a script, it either renders the script harmless before forwarding the request or response to its destination, or it blocks the connection.
- **HTML SQL Injection.** Examines requests that contain form field data for attempts to inject SQL commands into an SQL database. When this check detects injected SQL code, it either blocks the request or renders the injected SQL code harmless before forwarding the request to the Web server.

  Note: If both of the following conditions apply to your configuration, you should make certain that your Application Firewall is correctly configured:
  - If you enable the HTML Cross-Site Scripting check or the HTML SQL Injection check (or both), and
  - Your protected Web sites accept file uploads or contain Web forms that can contain large POST body data.

  For more information about configuring the Application Firewall to handle this case, see "Configuring the Application Firewall."
- **Buffer Overflow.** Examines requests to detect attempts to cause a buffer overflow on the Web server.
- **Cookie Consistency.** Examines cookies returned with user requests to verify that they match the cookies your Web server set for that user. If a modified cookie is found, it is stripped from the request before the request is forwarded to the Web server.

The Buffer Overflow check is simple; you can usually enable blocking for it immediately. The other three top-level checks are considerably more complex and require configuration before you can safely use them to block traffic. Citrix strongly recommends that, rather than attempting to configure these checks manually, you enable the learning feature and allow it to generate the necessary exceptions.

HTML Cross-Site Scripting Check

Jan 19, 2015

The HTML Cross-Site Scripting check examines both the headers and the POST bodies of user requests for possible cross-site scripting attacks. If it finds a cross-site script, it either modifies (transforms) the request to render the attack harmless, or blocks the request.

To prevent misuse of the scripts on your protected web sites to breach security on your web sites, the HTML Cross-Site Scripting check blocks scripts that violate the same origin rule, which states that scripts should not access or modify content on any server but the server on which they are located. Any script that violates the same origin rule is called a cross-site script, and the practice of using scripts to access or modify content on another server is called cross-site scripting. The reason cross-site scripting is a security issue is that a web server that allows cross-site scripting can be attacked with a script that is not on that web server, but on a different web server, such as one owned and controlled by the attacker.

Unfortunately, many companies have a large installed base of JavaScript-enhanced web content that violates the same origin rule. If you enable the HTML Cross-Site Scripting check on such a site, you have to generate the appropriate exceptions so that the check does not block legitimate activity.

If you use the wizard or the configuration utility, in the Modify HTML Cross-Site Scripting Check dialog box, on the General tab you can enable or disable the Block, Log, Learn, and Statistics actions, and in addition the following parameters:

- **Transform.** If enabled, the application firewall makes the following changes to requests that match the HTML Cross-Site Scripting check:
  - Left angle bracket (<) to HTML character entity equivalent (<)
  - Right angle bracket (>) to HTML character entity equivalent (>)

  This ensures that browsers do not interpret unsafe html tags, such as **<script>**, and thereby execute malicious code. If you enable both request-header checking and transformation, any special characters found in request headers are also modified as described above. If scripts on your protected web site contain cross-site scripting features, but your web site does not rely upon those scripts to operate correctly, you can safely disable blocking and enable transformation. This configuration ensures that no legitimate web traffic is blocked, while stopping any potential cross-site scripting attacks.
- **Check complete URLs.** If checking of complete URLs is enabled, the application firewall examines entire URLs for HTML cross-site scripting attacks instead of checking just the query portions of URLs.
- **Check Request headers.** If Request header checking is enabled, the application firewall examines the headers of requests for HTML cross-site scripting attacks, instead of just URLs. If you use the configuration utility, you can enable this parameter in the **Settings** tab of the application firewall profile.

If you use the command-line interface, you can enter the following commands to configure the HTML Cross-Site Scripting Check:

- set appfw profile <name> -crossSiteScriptingAction [**block**] [**learn**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -crossSiteScriptingTransformUnsafeHTML ([**ON**] | [**OFF**])
- set appfw profile <name> -crossSiteScriptingCheckCompleteURLs ([**ON**] | [**OFF**])
- set appfw profile <name> -checkRequestHeaders ([**ON**] | [**OFF**])

To specify relaxations for the HTML Cross-Site Scripting check, you must use the configuration utility. On the Checks tab of the Modify HTML Cross-Site Scripting Check dialog box, click Add to open the Add HTML Cross-Site Scripting Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify HTML Cross-Site Scripting Check Relaxation dialog box. Either dialog box provides the same options for configuring a relaxation, as described in "Manual Configuration By Using the Configuration Utility."

Following are examples of HTML Cross-Site Scripting check relaxations:

Web Form Field Expressions

- **Logon Fields.** The following expression exempts all fields beginning with the string `logon_` followed by a string of upper- and lower-case letters or numbers that is at least two characters long and no more than fifteen characters long:

  ```
  ^logon_[0-9A-Za-z]{2,15}$
  ```
- **Name Fields.** The following expression exempts form fields with names beginning with `Name_` followed by a string beginning with a letter or number and consisting of from one to twenty letters, numbers, or the apostrophe or hyphen symbol:

  ```
  ^Name_[0-9A-Za-z][0-9A-Za-z'-]{0,20}$
  ```
- **Name Fields (Special Characters).** If your web site has Turkish-speaking customers whose first names may contain special characters, you might have a form field that begins with the string Turkish-Name_ on their logon page. In addition, the customers may use the same special characters in their names. The special characters in both of these strings must be represented as encoded UTF-8 strings. The following expression exempts form fields beginning with Turkish-Name_ and containing Turkish special characters:

  ```
  ^T\xC3\xBCrk\xC3\xA7e-Name_([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])+$
  ```
- **Session-ID Fields.** The following expression exempts all fields beginning with the string `sessionid-` followed by a ten-digit number:

  ```
  ^sessionid-[0-9]{10,10}$
  ```

URL Expressions

- **URLs using JavaScript.** You can use a single expression to exempt all URLs that end with a filename that follows a specified pattern. The following expression exempts all URLs that end with the string `query_` followed by a string of upper- and lower-case letters or numbers that is at least two characters long and no more than forty characters long, and ending with the string `.js`:

  ```
  query_[0-9A-Za-z]{2,40}[.]js$
  ```
- **URLs containing a Specified String.** You can use an expression to exempt all URLs that contain a specific string. The following expression exempts all URLs that contain the string `prodinfo`:

  ```
  ^https?://(([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])(([0-9A-Za-z_-]|\\x[0-9A-Fa-f]
  [0-9A-Fa-f]+[.])+[a-z]{2,6}/[^<>?]*\?prodinfo[^<>?]*$
  ```
  In the above expression , each character class has been grouped with the string \\x[0-9A-Fa-f][0-9A-Fa-f], which matches all properly constructed character encoding strings but does not allow stray backslash characters that are not associated with a UTF-8 character encoding string. The double backslash (\\) is an escaped backslash, which tells the application firewall to interpret it as a literal backslash. If you included only one backslash, the application firewall would interpret the following left square bracket ([) as a literal character instead of as the opening of a character class, which would break the expression.

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the URL you want to add as an exception, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results you do not want or expect, such as blocking access to web content that you did not intend to block or allowing an attack that the HTML Cross-Site Scripting check would otherwise have blocked.

HTML SQL Injection Check

Jul 24, 2015

The HTML SQL Injection check provides special defenses against injection of unauthorized SQL code that might break security. It examines both the headers and the POST bodies of requests for injected SQL code. If the application firewall detects unauthorized SQL code in a user request, it either transforms the request, to render the SQL code inactive, or blocks the request.

Many web applications have web forms that use SQL to communicate with relational database servers. Often, the scripts that pass web form information to the database do not validate the information provided by the user before sending it to the database. Malicious code or a hacker can use the insecure web form to send SQL commands to the web server.

If you use the wizard or the configuration utility, in the Modify HTML SQL Injection Check dialog box, on the General tab you can enable or disable the Block, Log, Statistics, and Learn actions, and the following actions:

- **Transform.** Make the following changes to requests that match the HTML SQL Injection check:
  - Single straight quote (') to double straight quote ('')
  - Backslash (\) to double backslash (\\).
  - Semicolon (;) is dropped completely.

  These three characters (special strings) are necessary to issue commands to an SQL server. Unless an SQL command is prefaced with a special string, most SQL servers ignore that command. For this reason, the changes that the application firewall performs when transformation is enabled prevent an attacker from injecting active SQL. After these changes are made, it is safe to forward the request to your protected web site. When web forms on your protected web site may legitimately contain SQL special strings, but the web form does not rely upon the special strings to operate correctly, you can disable blocking and enable transformation to prevent blocking of legitimate web form data without reducing the protection that the application firewall provides to your protected web sites.

  Note: You normally enable either transformation or blocking, but not both. If you have blocking enabled, enabling transformation is redundant because the application firewall already blocks access to requests that contain injected SQL.
- **Restrict checks to fields containing SQL special characters.** If you configure the application firewall to check only fields that contain SQL special strings, the application firewall skips web form fields that do not contain special characters. Since most SQL servers do not process SQL commands that are not preceded by a special character, enabling this parameter can significantly reduce the load on the application firewall and speed up processing without placing your protected web sites at risk.

- **SQL comments handling.** By default, the application firewall checks all SQL comments for injected SQL commands. Many SQL servers ignore anything in a comment, however, even if it is preceded by an SQL special character. For faster processing, if your SQL server ignores comments, you can configure the application firewall to skip comments when examining requests for injected SQL. The SQL comments handling options are:
  - **ANSI.** Skip ANSI-format SQL comments, which are normally used by UNIX-based SQL databases.
  - **Nested.** Skip nested SQL comments, which are normally used by Microsoft SQL Server.
  - **ANSI/Nested.** Skip comments that adhere to both the ANSI and nested SQL comment standards. Comments that match only the ANSI standard, or only the nested standard, are still checked for injected SQL.
  - **Check all Comments.** Check the entire request for injected SQL, without skipping anything. The default setting.

  Caution: In most cases, you should not choose the Nested or the ANSI/Nested option unless your back-end database runs on Microsoft SQL Server. Most other types of SQL server software do not

recognize nested comments. If nested comments appear in a request directed to another type of SQL server, they may indicate an attempt to breach security on that server.

- **Check Request headers.** Examine the headers of requests for HTML SQL Injection attacks, instead of just URLs. If you use the configuration utility, you can enable this parameter in the **Settings** tab of the application firewall profile.

  Caution: If you enable both request header checking and transformation, any SQL special characters found in headers are also transformed. The Accept, Accept-Charset, Accept-Encoding, Accept-Language, Expect, and User-Agent headers normally contain semicolons (;), so enabling both Request header checking and transformation simultaneously may cause errors.

If you use the command-line interface, you can enter the following commands to configure the HTML SQL Injection Check:

- set appfw profile <name> -SQLInjectionAction [**block**] [**learn**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -SQLInjectionTransformSpecialChars ([**ON**] | [**OFF**])
- set appfw profile <name> -SQLInjectionOnlyCheckFieldsWithSQLChars ([**ON**]|[**OFF**])
- set appfw profile <name> -SQLInjectionParseComments ([**checkall**] | [**ansi**|**nested**] | [**ansinested**])

To specify relaxations for the HTML SQL Injection check, you must use the configuration utility. On the Checks tab of the Modify HTML SQL Injection Check dialog box, click Add to open the Add HTML SQL Injection Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify HTML SQL Injection Check Relaxation dialog box. Either dialog box provides the same options for configuring a relaxation, as described in "Manual Configuration By Using the Configuration Utility."

Following are examples of HTML SQL Injection check relaxations:

Web Form Field Name Expressions

- **Logon Fields.** The following expression exempts all fields beginning with the string **logon_** followed by a string of letters or numbers that is at least two characters long and no more than fifteen characters long:

  ```
  ^logon_[0-9A-Za-z]{2,15}$
  ```
- **Name Fields.** The following expression exempts form fields with names beginning with **Name_** followed by a string beginning with a letter or number and consisting of from one to twenty letters, numbers, or the apostrophe or hyphen symbol:

  ```
  ^Name_[0-9A-Za-z][0-9A-Za-z'-]{0,20}$
  ```
- **Name Fields (Special Characters).** If your web site has Turkish-speaking customers whose first names may contain special characters, you might have a form field that begins with the string Turkish-Name_ on their logon page. In addition, the customers may use the same special characters in their names. The special characters in both of these strings must be represented as encoded UTF-8 strings. The following expression exempts form fields beginning with Turkish-Name_ and containing Turkish special characters:

  ```
  ^T\xC3\xBCrk\xC3\xA7e-Name_([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])+$
  ```
- **Session-ID Fields.** The following expression exempts all fields beginning with the string **sessionid-** followed by a ten-digit number:

  ```
  ^sessionid-[0-9]{10,10}$
  ```

Action URL Expressions

- **URLs using JavaScript.** You can use a single expression to exempt all URLs that end with a filename that follows a specified pattern. The following expression exempts all URLs that end with the string query_ followed by a string of upper- and lower-case letters or numbers that is at least two characters long and no more than forty characters long, and that end with the string .js:

  ```
  query_[0-9A-Za-z]{2,40}[.]js$
  ```

- **URLs containing a Specified String.** You can use an expression to exempt all URLs that contain a specific string. The following expression exempts all URLs that contain the string prodinfo:

  ```
  ^https?://(([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])(([0-9A-Za-z_-]|\\x[0-9A-Fa-f]
  [0-9A-Fa-f]+[.])+[a-z]{2,6}/[^<>?]*\?prodinfo[^<>?]*$
  ```

  In the expression above, each character class has been grouped with the string \\x[0-9A-Fa-f][0-9A-Fa-f], which matches all properly constructed character encoding strings but does not allow stray backslash characters that are not associated with a UTF-8 character encoding string. The double backslash (\\) is an escaped backslash, which tells the application firewall to interpret it as a literal backslash. If you included only one backslash, the application firewall would interpret the following left square bracket ([) as a literal character instead of as the opening of a character class, which would break the expression.

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the URL that you want to add as an exception, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results that you do not want, such as blocking access to web content that you did not intend to block or allowing an attack that the HTML SQL Injection check would otherwise have blocked.

Buffer Overflow Check

Feb 03, 2014

The Buffer Overflow check detects attempts to cause a buffer overflow on the web server. If the application firewall detects that the URL, cookies or header are longer than the specified maximum length in a request, it blocks that request because it might be an attempt to cause a buffer overflow.

The Buffer Overflow check prevents attacks against insecure operating-system or web-server software that can crash or behave unpredictably when it receives a data string that is larger than it can handle. Proper programming techniques prevent buffer overflows by checking incoming data and either rejecting or truncating overlong strings. Many programs, however, do not check all incoming data and are therefore vulnerable to buffer overflows. This issue especially affects older versions of web-server software and operating systems, many of which are still in use.

If you use the wizard or the configuration utility, in the Modify Buffer Overflow Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions. On the Checks tab, you can set the following parameters:

- **Maximum URL Length.** The maximum length the application firewall allows in a requested URL. Requests with longer URLs are blocked. Possible Values: 0-65536. Default: 1024
- **Maximum Cookie Length.** The maximum length the application firewall allows for all cookies in a request. Excess cookies are stripped from requests before those requests are forwarded to your protected web server. Possible Values: 0-65536. Default: 4096
- **Maximum Header Length.** The maximum length the application firewall allows for HTTP headers. Requests with longer headers are blocked. Possible Values: 0-65536. Default: 4096

If you use the command-line interface, you can add the following Buffer Overflow Check arguments to the set appfwl profile <profileName> command:

- -bufferOverflowAction [**block**] [**log** | [**stats**]
- -bufferOverflowMaxURLLength <positiveInteger>
- -bufferOverflowMaxCookieLength <positiveInteger>
- -bufferOverflowMaxHeaderLength <positiveInteger>

Cookie Consistency Check

Feb 03, 2014

The Cookie Consistency check examines cookies returned by users, to verify that they match the cookies that your web site set for that user. If a modified cookie is found, it is stripped from the request before the request is forwarded to the web server. You can also configure the Cookie Consistency check to transform all of the server cookies that it processes, by encrypting the cookies, proxying the cookies, or adding flags to the cookies. This check applies to requests and responses.

An attacker would normally modify a cookie to gain access to sensitive private information by posing as a previously authenticated user, or to cause a buffer overflow. The Buffer Overflow check protects against attempts to cause a buffer overflow by using a very long cookie. The Cookie Consistency check focuses on the first scenario.

If you use the wizard or the configuration utility, in the Modify Cookie Consistency Check dialog box, on the General tab you can enable or disable the following actions:

- Block
- Log
- Learn
- Statistics
- Transform. If enabled, the Transform action modifies all cookies as specified in the following settings:
  - **Encrypt Server Cookies.** Encrypt cookies set by your web server, except for any listed in the Cookie Consistency check relaxation list, before forwarding the response to the client. Encrypted cookies are decrypted when the client sends a subsequent request, and the decrypted cookies are reinserted into the request before it is forwarded to the protected web server. Specify one of the following types of encryption:
    - **None.** Do not encrypt or decrypt cookies. The default.
    - **Decrypt only.** Decrypt encrypted cookies only. Do not encrypt cookies.
    - **Encrypt session only.** Encrypt session cookies only. Do not encrypt persistent cookies. Decrypt any encrypted cookies.
    - **Encrypt all.** Encrypt both session and persistent cookies. Decrypt any encrypted cookies.
      Note: When encrypting cookies, the application firewall adds the **HttpOnly** flag to the cookie. This flag prevents scripts from accessing and parsing the cookie. The flag therefore prevents a script-based virus or trojan from accessing a decrypted cookie and using that information to breach security. This is done regardless of the Flags to Add in Cookies parameter settings, which are handled independently of the Encrypt Server Cookies parameter settings.
- **Proxy Server Cookies.** Proxy all non-persistent (session) cookies set by your web server, except for any listed in the Cookie Consistency check relaxation list. Cookies are proxied by using the existing application firewall session cookie. The application firewall strips session cookies set by the protected web server and saves them locally before forwarding the response to the client. When the client sends a subsequent request, the application firewall reinserts the session cookies into the request before forwarding it to the protected web server. Specify one of the following settings:
  - **None.** Do not proxy cookies. The default.
  - **Session only.** Proxy session cookies only. Do not proxy persistent cookies.
    Note: If you disable cookie proxying after having enabled it (set this value to None after it was set to Session only), cookie proxying is maintained for sessions that were established before you disabled it. You can therefore safely disable this feature while the application firewall is processing user sessions.
- **Flags to Add in Cookies**. Add flags to cookies during transformation. Specify one of the following settings:
  - **None.** Do not add flags to cookies. The default.
  - **HTTP only.** Add the HttpOnly flag to all cookies. Browsers that support the HttpOnly flag do not allow scripts to

access cookies that have this flag set.
- **Secure**. Add the Secure flag to cookies that are to be sent only over an SSL connection. Browsers that support the Secure flag do not send the flagged cookies over an insecure connection.
- **All.** Add the HttpOnly flag to all cookies, and the Secure flag to cookies that are to be sent only over an SSL connection.

If you use the command-line interface, you can enter the following commands to configure the Cookie Consistency Check:

- set appfw profile <name> -cookieConsistencyAction [**block**] [**learn**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -cookieTransforms ([**ON**] | [**OFF**])
- set appfw profile <name> -cookieEncryption ([**none**] | [**decryptOnly**] | [**encryptSession**] | [**encryptAll**])
- set appfw profile <name> -cookieProxying ([**none**] | [**sessionOnly**])
- set appfw profile <name> -addCookieFlags ([**none**] | [**httpOnly**] | [**secure**] | [**all**])

To specify relaxations for the Cookie Consistency check, you must use the configuration utility. On the Checks tab of the Modify Cookie Consistency Check dialog box, click Add to open the Add Cookie Consistency Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify Cookie Consistency Check Relaxation dialog box. Either dialog box provides the same options for configuring a relaxation.

Following are examples of Cookie Consistency check relaxations:

- **Logon Fields.** The following expression exempts all form fields beginning with the string **logon_** followed by a string of letters or numbers that is at least two characters long and no more than fifteen characters long:

  ```
  ^logon_[0-9A-Za-z]{2,15}$
  ```
- **Logon Fields (special characters).** The following expression exempts all form fields beginning with the string **türkçe-logon_** followed by a string of letters or numbers that is at least two characters long and no more than fifteen characters long:

  ```
  ^t\xC3\xBCrk\xC3\xA7e-logon_[0-9A-Za-z]{2,15}$
  ```
- **Arbitrary strings.** Allow cookies that contain the string **sc-item_**, followed by the ID of an item that the user has added to his shopping cart ([0-9A-Za-z]+), a second underscore (_), and finally the number of these items he wants ([1-9][0-9]?), to be user-modifiable:

  ```
  ^sc-item_[0-9A-Za-z]+_[1-9][0-9]?$
  ```

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the URL you want to add as an exception, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results you do not want or expect, such as blocking access to web content that you did not intend to block or allowing an attack that the Cookie Consistency check would otherwise have blocked.

Data Leak Prevention Checks

Mar 28, 2012

The data-leak-prevention checks filter responses to prevent leaks of sensitive information, such as credit card numbers and social security numbers, to unauthorized recipients.

Credit Card Check

Feb 03, 2014

The Credit Card check provides special handling for credit card numbers. A web application does not usually send a credit card number in a response to a user request, even when the user supplies a credit card number in the request. The application firewall examines web server responses, including headers, for credit card numbers. If it finds a credit card number in the response, and the administrator has not configured it to allow credit card numbers to be sent, it responds in one of two ways:

- It blocks the response.
- It replaces all but the final group of digits in the credit card with x's. For example, a credit card number of 9876-5432-1234-5678 would be rendered xxxx-xxxx-xxxx-5678.

The Credit Card check prevents attackers from exploiting a security flaw in your web server software or on your web site to obtain credit card numbers of your customers. If your web sites do not have access to credit card information, you do not need to configure this check. If you have a shopping cart or other application that can access credit card numbers, or your web sites have access to database servers that contain credit card numbers, you should configure protection for each type of credit card that you accept.

Note: A web site that does not access an SQL database usually does not have access to sensitive private information such as credit card numbers.

If you use the wizard or the configuration utility, in the Credit Card Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions, and the following actions:

- **X-Out.** Mask any credit card number detected in a response by replacing each digit, except the digits in the final group, with the letter "X."
- **Maximum credit cards allowed per page.** Allow up to the specified number of credit card numbers per page in responses without masking the credit card numbers or blocking the response. The Maximum is set to zero (0) by default. Web pages do not usually contain unmasked credit card numbers, but occasionally a web page might legitimately contain a credit card number or even a list of credit card numbers. To allow one or more credit card numbers to appear in a web page before masking the numbers or blocking the response, change the value in the "Maximum credit cards allowed per page" text box to the number of credit cards that you want to allow.

To configure the types of credit cards to be protected, in the Modify Credit Card Check dialog box, select each credit card type that you want to protect, and then click Protect. If you want to cancel protection for a credit card type, select that credit card type and then click Unprotect. You can hold down your Shift or Ctrl key while choosing credit card types, and then enable or disable several credit card types at once by clicking the Protect or Unprotect button while multiple credit card types are selected.

If you use the command-line interface, you can enter the following commands to configure the Credit Card Check:

- set appfw profile <name> -creditCardAction [**block**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -creditCard (**VISA** | **MASTERCARD** | **DISCOVER** | **AMEX** | **JCB** | **DINERSCLUB**)
- set appfw profile <name> -creditCardMaxAllowed <integer>
- set appfw profile <name> -creditCardXOut ([**ON**] | [**OFF**])

Safe Object Check

Feb 03, 2014

The Safe Object check provides user-configurable protection for sensitive business information, such as customer numbers, order numbers, and country-specific or region-specific telephone numbers or postal codes. A user-defined regular expression or custom plug-in tells the application firewall the format of this information and defines the rules to be used to protect it. If a string in a user request matches a safe object definition, the application firewall either blocks the response, masks the protected information, or removes the protected information from the response before sending it to the user, depending on how you configured that particular safe object rule.

The Safe Object check prevents attackers from exploiting a security flaw in your web server software or on your web site to obtain sensitive private information, such as company credit card numbers or social security numbers. If your web sites do not have access to these types of information, you do not need to configure this check. If you have a shopping cart or other application that can access such information, or your web sites have access to database servers that contain such information, you should configure protection for each type of sensitive private information that you handle and store.

Note: A web site that does not access an SQL database usually does not have access to sensitive private information.
The Safe Object Check dialog box is unlike that for any other check. Each safe object expression that you create is the equivalent of a separate security check, similar to the Credit Card check, for that type of information. If you use the wizard or the configuration utility, you add a new expression by clicking Add and configuring the expression in the Add Safe Object dialog box. You modify an existing expression by selecting it, then clicking Open, and then configuring the expression in the Modify Safe Object dialog box.

In the Safe Object dialog box for each safe object expression, you can configure the following:

- **Safe Object Name.** A name for your new safe object. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 255 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols.
- **Actions.** Enable or disable the Block, Log, and Statistics actions, and the following actions:
  - **X-Out.** Mask any information that matches the safe object expression with the letter “X”.
  - **Remove.** Remove any information that matches the safe object expression.
- **Regular Expression.** Enter a PCRE-compatible regular expression that defines the safe object. You can create the regular expression in one of three ways: by typing the regular expression directly into the text box, by using the **Regex Tokens** menu to enter regular expression elements and symbols directly into the text box, or by opening the Regular Expressions Editor and using it to construct the expression. The regular expression must consist of ASCII characters only. Do not cut and paste characters that are not part of the basic 128-character ASCII set. If you want to include non-ASCII characters, you must manually type those characters in PCRE hexadecimal character encoding format.
  Note: Do not use start anchors (^) at the beginning of Safe Object expressions, or end anchors ($) at the end of Safe Object expressions. These PCRE entities are not supported in Safe Object expressions, and if used, will cause your expression not to match what it was intended to match.
- **Maximum Match Length.** Enter a positive integer that represents the maximum length of the string that you want to match. For example, if you want to match U.S. social security numbers, enter the number eleven (11) in this field. That allows your regular expression to match a string with nine numerals and two hyphens. If you want to match California driver’s license numbers, enter the number eight (8).
  Caution: If you do not enter a maximum match length in this field, the application firewall uses a default value of one (1) when filtering for strings that match your safe object expressions. As a result, most safe object expressions fail to match their target strings.

You cannot use the command-line interface to configure the Safe Object check. You must configure it by using either the application firewall wizard or the configuration utility.

Following are examples of Safe Object check regular expressions:

- Look for strings that appear to be U.S. social security numbers, which consist of three numerals (the first of which must not be zero), followed by a hyphen, followed by two more numerals, followed by a second hyphen, and ending with a string of four more numerals:
  `[1-9][0-9]{2,2}-[0-9]{2,2}-[0-9]{4,4}`
- Look for strings that appear to be California driver's license IDs, which start with a letter and are followed by a string of exactly seven numerals:
  `[A-Za-z][0-9]{7,7}`
- Look for strings that appear to be Example Manufacturing customer IDs which, consist of a string of five hexadecimal characters (all the numerals and the letters A through F), followed by a hyphen, followed by a three-letter code, followed by a second hyphen, and ending with a string of ten numerals:
  `[0-9A-Fa-f]{5,5}-[A-Za-z]{3,3}-[0-9]{10,10}`

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write to ensure that they define exactly the type of string you want to add as a safe object definition, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results you did not want or expect, such as blocking access to web content that you did not intend to block.

Advanced Form Protection Checks

Mar 28, 2012

The advanced Form Protection checks examine web form data to prevent attackers from compromising your system by modifying the web forms on your web sites or sending unexpected types and quantities of data to your web site in a form.

Field Formats Check

Feb 05, 2014

The Field Formats check verifies the data that users send to your web sites in a web form. It examines both the length and type of data to ensure that it is appropriate for the form field in which it appears. If the application firewall detects inappropriate web form data in a user request, it blocks the request. This check applies to HTML requests only. It does not apply to XML requests.

By preventing an attacker from sending inappropriate web form data to your web site, the Field Formats check prevents certain types of attacks on your web site and database servers. For example, if a particular field expects the user to enter a phone number, the Field Formats check examines the user's response to ensure that the data matches the format for a phone number. If a particular field expects a first name, the Field Formats check ensures that the data in that field is of a type and length appropriate for a first name. It does the same thing for each form field that you configure it to protect.

The Field Formats check provides a different type of protection than does the Form Field Consistency check. The Form Field Consistency check verifies that the structure of the web forms returned by users is intact, that data format restrictions configured in the HTML are respected, and that data in hidden fields has not been modified. It can do this without any specific knowledge about your web forms other than what it derives from the web form itself. The Field Formats check verifies that the data in each form field matches the specific formatting restrictions that you configured manually, or that the learning feature generated and you approved. In other words, the Form Field Consistency check enforces general web form security, while the Field Formats check enforces the specific rules that you set for your web forms.

Before it can protect your web forms, the Field Formats check requires that you configure the application firewall to recognize the type and length of data expected in each form field on each web form that you want to protect.

If you use the wizard or the configuration utility, in the Modify Field Formats Check dialog box, on the General tab you can enable or disable the Block, Log, Learn, and Statistics actions and the following parameters:

- **Field Type.** Assign a default field type to form fields in web forms that do not have a field type. This parameter is not set by default. You can assign any field type that is defined on your application firewall as the default field type. Caution: If you set a restrictive default field type and do not disable blocking until you are certain that the field types assigned to your form fields are correct, users may be unable to use your web forms.
- **Minimum Length.** The default minimum data length assigned to form fields in web forms that do not have an explicit setting. This parameter is set to 0 by default, which allows the user to leave the field blank. Any higher setting forces users to fill in the field.
- **Maximum Length.** The default maximum data length assigned to form fields in web forms that do not have an explicit setting. This parameter is set to 65535 by default.

If you use the command-line interface, you can enter the following commands to configure the Field Formats Check:

- set appfw profile <name> -fieldFormatAction [**block**] [**learn**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -defaultFieldFormatType <string>
- set appfw profile <name> -defaultFieldFormatMinLength <integer>
- set appfw profile <name> -defaultFieldFormatMaxLength <integer>

To specify relaxations for the Field Formats check, you must use the configuration utility. On the Checks tab of the Modify Field Formats Check dialog box, click Add to open the Add Field Formats Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify Field Formats Check Relaxation dialog box. Either dialog box provides the

same options for configuring a relaxation.

Following are examples of Field Formats check relaxations:

- Choose form fields with the name FirstName:

  ```
  ^FirstName$
  ```
- Choose form fields with names that begin with Name_ and are followed by a string beginning with a letter or number and consisting of from one to twenty letters, numbers, or the apostrophe or hyphen symbol:

  ```
  ^Name_[0-9A-Za-z][0-9A-Za-z'-]{0,20}$
  ```
- Choose form fields with names that begin with Turkish-FirstName_ and are otherwise the same as the previous expression, except that they can contain Turkish special characters throughout:

  ```
  ^T\xC3\xBCrk\xC3\xA7e-FirstName_([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])+$
  ```
- Choose form field names that begin with a letter or number, consist of a combination of letters and/or numbers only, and that contain the string Num anywhere in the string:

  ```
  ^[0-9A-Za-z]*Num[0-9A-Za-z]*$
  ```

Form Field Consistency Check

Feb 05, 2014

The Form Field Consistency check examines the web forms returned by users of your web site, and verifies that web forms were not modified inappropriately by the client. This check applies only to HTML requests that contain a web form, with or without data. It does not apply to XML requests.

The Form Field Consistency check prevents clients from making unauthorized changes to the structure of the web forms on your web site when they fill out and submit a form. It also ensures that the data a user submits meets the HTML restrictions for length and type, and that data in hidden fields is not modified. This prevents an attacker from tampering with a web form and using the modified form to gain unauthorized access to web site, redirect the output of a contact form that uses an insecure script and thereby send unsolicited bulk email, or exploit a vulnerability in your web server software to gain control of the web server or the underlying operating system. Web forms are a weak link on many web sites and attract a wide range of attacks.

The Form Field Consistency check verifies all of the following:

- If a field is sent to the user, the check ensures that it is returned by the user.
- The check enforces HTML field lengths and types.
  Note: The Form Field Consistency check enforces HTML restrictions on data type and length but does not otherwise validate the data in web forms. You can use the Field Formats check to set up rules that validate data returned in specific form fields on your web forms.
- If your web server does not send a field to the user, the check does not allow the user to add that field and return data in it.
- If a field is a read-only or hidden field, the check verifies that the data has not changed.
- If a field is a list box or radio button field, the check verifies that the data in the response corresponds to one of the values in that field.

If a web form returned by a user violates one or more of the Form Field consistency checks, and you have not configured the application firewall to allow that web form to violate the Form Field Consistency checks, the request is blocked.

If you use the wizard or the configuration utility, in the Modify Form Field Consistency Check dialog box, on the General tab you can enable or disable the Block, Log, Learn, and Statistics actions.

You also configure Sessionless Field Consistency in the General tab. If Sessionless Field Consistency is enabled, the application firewall checks only the web form structure, dispensing with those parts of the Form Field Consistency check that depend upon maintaining session information. This can speed the Form Field Consistency check with little security penalty for web sites that use many forms. To use Sessionless Field Consistency on all web forms, select On. To use it only for forms submitted with the HTTP POST method, select postOnly

If you use the command-line interface, you can enter the following command to configure the Form Field Consistency Check:

- set appfw profile <name> -fieldConsistencyAction [**block**] [**learn**] [**log**] [**stats**] [**none**]

To specify relaxations for the Form Field Consistency check, you must use the configuration utility. On the Checks tab of the Modify Form Field Consistency Check dialog box, click Add to open the Add Form Field Consistency Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify Form Field Consistency Check Relaxation

dialog box. Either dialog box provides the same options for configuring a relaxation, as described in "Manual Configuration By Using the Configuration Utility."

Following are examples of Form Field Consistency check relaxations:

Form Field Names

- Choose form fields with the name UserType:

  ```
  ^UserType$
  ```
- Choose form fields with names that begin with UserType_ and are followed by a string that begins with a letter or number and consists of from one to twenty-one letters, numbers, or the apostrophe or hyphen symbol:

  ```
  ^UserType_[0-9A-Za-z][0-9A-Za-z'-]{0,20}$
  ```
- Choose form fields with names that begin with Turkish-UserType_ and are otherwise the same as the previous expression, except that they can contain Turkish special characters throughout:

  ```
  ^T\xC3\xBCrk\xC3\xA7e-UserType_([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])+$
  ```
  Note: See "PCRE Character Encoding Format" for a complete description of supported special characters and how to encode them properly.
- Choose form field names that begin with a letter or number, consist of a combination of letters and/or numbers only, and that contain the string Num anywhere in the string:

  ```
  ^[0-9A-Za-z]*Num[0-9A-Za-z]*$
  ```

Form Field Action URLs

- Choose URLs beginning with http://www.example.com/search.pl? and containing any string after the query except for a new query:

  ```
  ^http://www[.]example[.]com/search[.]pl\?[^?]*$
  ```
- Choose URLs that begin with http://www.example-español.com and have paths and filenames that consist of upper-case and lower-case letters, numbers, non-ASCII special characters, and selected symbols in the path. The ñ character and any other special characters are represented as encoded UTF-8 strings containing the hexadecimal code assigned to each special character in the UTF-8 charset:

  ```
  ^http://www[.]example-espa\xC3\xB1ol[.]com/(([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])
  ([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*/)*([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])
  ([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*[.](asp|htp|php|s?html?)$
  ```
- Choose all URLs that contain the string /search.cgi?:

  ```
  ^[^?<>]*/search[.]cgi\?[^?<>]*$
  ```

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the URL you want to add as an exception, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results you do not want or expect, such as blocking access to web content that you did not intend to block or allowing an attack that the Cookie Consistency check would otherwise have blocked.

CSRF Form Tagging Check

Feb 05, 2014

The Cross Site Request Forgery (CSRF) Form Tagging check tags each web form sent by a protected web site to users with a unique and unpredictable FormID, and then examines the web forms returned by users to ensure that the supplied FormID is correct. This check protects against cross-site request forgery attacks. This check applies only to HTML requests that contain a web form, with or without data. It does not apply to XML requests.

The CSRF Form Tagging check prevents attackers from using their own web forms to send high volume form responses with data to your protected web sites. This check requires relatively little CPU processing capacity compared to certain other security checks that analyze web forms in depth. It is therefore able to handle high volume attacks without seriously degrading the performance of the protected web site or the application firewall itself.

Before you enable the CSRF Form Tagging check, you should be aware of the following:

- You need to enable form tagging. The CSRF check depends on form tagging and does not work without it.
- You should disable the Citrix NetScaler Integrated Caching feature for all web pages containing forms that are protected by that profile. The Integrated Caching feature and CSRF form tagging are not compatible.
- You should consider enabling Referer checking. Referer checking is part of the Start URL check, but it prevents cross-site request forgeries, not Start URL violations. Referer checking also puts less load on the CPU than does the CSRF Form Tagging check. If a request violates Referer checking, it is immediately blocked, so the CSRF Form Tagging check is not invoked.
- The CSRF Form Tagging check does not work with web forms that use different domains in the form-origin URL and form-action URL. For example, CSRF Form Tagging cannot protect a web form with a form-origin URL of `http://www.example.com/` and a form action URL of `http://www.example.org/form.pl`, because `example.com` and `example.org` are different domains.

If you use the wizard or the configuration utility, in the Modify CSRF Form Tagging Check dialog box, on the General tab you can enable or disable the Block, Log, Learn and Statistics actions.

If you use the command-line interface, you can enter the following command to configure the CSRF Form Tagging Check:

- set appfw profile <name> -fieldConsistencyAction [**block**] [**log**] [**learn**] [**stats**] [**none**]

To specify relaxations for the CSRF Form Tagging check, you must use the configuration utility. On the Checks tab of the Modify CSRF Form Tagging Check dialog box, click Add to open the Add CSRF Form Tagging Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify CSRF Form Tagging Check Relaxation dialog box. Either dialog box provides the same options for configuring a relaxation.

Following are examples of CSRF Form Tagging check relaxations:

Note: The following expressions are URL expressions that can be used in both the Form Origin URL and Form Action URL roles.

- Choose URLs beginning with http://www.example.com/search.pl? and containing any string after the query, except for a new query:

  `^http://www[.]example[.]com/search[.]pl\?[^?]*$`
- Choose URLs that begin with http://www.example-español.com and have paths and filenames that consist of upper-case and lower-case letters, numbers, non-ASCII special characters, and selected symbols in the path. The ñ character

and any other special characters are represented as encoded UTF-8 strings containing the hexadecimal code assigned to each special character in the UTF-8 charset:

```
^http://www[.]example-espa\xC3\xB1ol[.]com/(([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])
([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*/)*([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])
([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*[.](asp|htp|php|s?html?)$
```

- Choose all URLs that contain the string /search.cgi?:

```
^[^?<>]*/search[.]cgi\?[^?<>]*$
```

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the URL that you want to add as an exception, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results you do not want, such as blocking access to web content that you did not intend to block or allowing an attack that the check would otherwise have blocked.

Managing CSRF Form Tagging Check Relaxations

Sep 03, 2013

You configure an exception (or relaxation) to the CSRF Form Tagging security check in the Add Cross-Site Request Forgery Tagging Check Relaxation dialog box or the Modify Cross-Site Request Forgery Tagging Check Relaxation dialog box.

1. Navigate to Security > Application Firewall > Profiles.
2. In the Profiles pane, select the profile you want to configure, and then click Open.
3. In the Configure Application Firewall Profile dialog box, click the Security Checks tab. The Security Checks tab contains the list of application firewall security checks.
4. In the Security Checks window, click CSRF Form Tagging, and then click Open. The Modify Cross-Site Request Forgery Tagging Check dialog box is displayed, with the Checks tab selected. The Checks tab contains a list of existing CSRF relaxations. The list might be empty if you have not either manually added any relaxations or approved any relaxations that were recommended by the learning engine. Beneath the list is a row of buttons that allow you to add, modify, delete, enable, or disable the relaxations on the list.
5. To add or modify a CSRF relaxation, do one of the following:
   - To add a new relaxation, click Add.
   - To modify an existing relaxation, select the relaxation that you want to modify, and then click Open.

   The Add Cross-Site Request Forgery Tagging Check Relaxation or Modify Cross-Site Request Forgery Tagging Check Relaxation dialog box is displayed. Except for the title, these dialog boxes are identical.
6. Fill in the dialog box as described below.
   - **Enabled check box**—Select to place this relaxation or rule in active use; clear to deactivate it.
   - **Form Origin URL**—In the text area, enter a PCRE-format regular expression that defines the URL that hosts the form.
   - **Form Action URL**—In the text area, enter a PCRE-format regular expression that defines the URL to which data entered into the form is delivered.
   - **Comments**—In the text area, type a comment. Optional.

   Note: For any element that requires a regular expression, you can type the regular expression, use the Regex Tokens menu to insert regular expression elements and symbols directly into the text box, or click Regex Editor to open the Add Regular Expression dialog box, and use it to construct the expression.
7. Click OK. The Add Cross-Site Request Forgery Tagging Check Relaxation or Modify Cross-Site Request Forgery Tagging Check Relaxation dialog box closes and you return to the Modify Cross-Site Request Forgery Tagging Check dialog box.
8. To remove a relaxation or rule, select it, and then click Remove.
9. To enable a relaxation or rule, select it, and then click Enable.
10. To disable a relaxation or rule, select it, and then click Disable.
11. To configure the settings and relationships of all existing relaxations in an integrated interactive graphic display, click Visualizer, and use the display tools.
12. To review and configure learned rules for the CSRF check, click Learning and perform the steps in "To configure and use the Learning feature."
13. Click OK.

URL Protection Checks

Mar 28, 2012

The URL Protection checks examine request URLs to prevent attackers from aggressively attempting to access multiple URLs (forceful browsing) or using a URL to trigger a known security vulnerability in web server software or web site scripts.

Start URL Check

Feb 03, 2014

The Start URL check examines the URLs in incoming requests and blocks the connection attempt if the URL does not meet the specified criteria. To meet the criteria, the URL must match an entry in the Start URL list, unless the Enforce URL Closure parameter is enabled. If you enable this parameter, a user who clicks a link on your Web site is connected to the target of that link.

The primary purpose of the Start URL check is to prevent repeated attempts to access random URLs on a Web site, (forceful browsing). Forceful browsing can be used to trigger a buffer overflow, find content that users were not intended to access directly, or find a back door into secure areas of your Web server.

If you use the wizard or the configuration utility, in the Modify Start URL Check dialog box, on the General tab you can enable or disable Block, Log, Statistics, Learn actions, and the following parameters:

- **Enforce URL Closure.** Allow users to access any web page on your web site by clicking a hyperlink on any other page on your web site. Users can navigate to any page on your web site that can be reached from the home page or any designated start page by clicking hyperlinks.
  Note: The URL closure feature allows any query string to be appended to and sent with the action URL of a web form submitted by using the HTTP GET method. If your protected web sites use forms to access an SQL database, make sure that you have the SQL injection check enabled and properly configured.
- **Sessionless URL Closure.** From the client's point of view, this type of URL closure functions in exactly the same way as standard, session-aware URL Closure, but uses a token embedded in the URL instead of a cookie to track the user's activity, which consumes considerably fewer resources.
  Note: When enabling sessionless (Sessionless URL Closure), you must also enable regular URL closure (Enforce URL Closure) or sessionless URL closure does not work.
- **Validate Referer Header.** Verify that the Referer header in a request that contains web form data from your protected web site instead of another web site. This action verifies that your web site, not an outside attacker, is the source of the web form. Doing so protects against cross-site request forgeries (CSRF) without requiring form tagging, which is more CPU-intensive than header checks. The application firewall can handle the HTTP Referer header in one of the following three ways, depending on which option you select in the drop-down list:
  - **Off.** Do not validate the Referer header.
  - **If-Present.** Validate the Referer header if a Referer header exists. If an invalid Referer header is found, the request generates a referer-header violation. If no Referer header exists, the request does not generate a referer-header violation. This option enables the application firewall to perform Referer header validation on requests that contain a Referer header, but not block requests from users whose browsers do not set the Referer header or who use web proxies or filters that remove that header.
  - **Always.** Always validate the Referer header. If there is no Referer header, or if the Referer header is invalid, the request generates a referer-header violation.

  Note: Although the referer header check and Start URL security check share the same action settings, it is possible to violate the referer header check without violating the Start URL check. The difference is visible in the logs, which log referer header check violations separately from Start URL check violations.

One Start URL setting, Exempt Closure URLs from Security Checks, is not configured in the Modify Start URL Check dialog box, but in the Settings tab of the Configure Application Firewall Profile dialog box. If enabled, this setting directs the application firewall not to run further security checks on URLs that meet the URL Closure criteria.

If you use the command-line interface, you can enter the following commands to configure the Start URL Check:

- set appfw profile <name> -startURLAction [**block**] [**learn**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -startURLClosure ([**ON**] | [**OFF**])
- set appfw profile <name> -sessionlessURLClosure ([**ON**] | [**OFF**])
- set appfw profile <name> -exemptClosureURLsFromSecurityChecks ([**ON**] | [**OFF**])
- set appfw profile <name> -RefererHeaderCheck ([**none**] | [**if-present**] | [**always**])

To specify relaxations for the Start URL check, you must use the configuration utility. On the Checks tab of the Modify Start URL Check dialog box, click Add to open the Add Start URL Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify Start URL Check Relaxation dialog box. Either dialog box provides the same options for configuring a relaxation.

Following are examples of Start URL check relaxations:

- Allow users to access the home page at www.example.com:

^http://www[.]example[.]com$

- Allow users to access all static HTML (.htm and .html), server-parsed HTML (.htp and .shtml), PHP (.php), and Microsoft ASP (.asp) format web pages at www.example.com:

  ^http://www[.]example[.]com/([0-9A-Za-z][0-9A-Za-z_-]*/)*
  [0-9A-Za-z][0-9A-Za-z_.-]*[.](asp|htp|php|s?html?)$
- Allow users to access web pages with pathnames or file names that contain non-ASCII characters at www.example-español.com:

  ^http://www[.]example-espa\xC3\xB1ol[.]com/(([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*/)*
  ([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*[.](asp|htp|php|s?html?)$
  Note: In the above expression, each character class has been grouped with the string \\x[0-9A-Fa-f][0-9A-Fa-f], which matches all properly-constructed character encoding strings but does not allow stray backslash characters that are not associated with a UTF-8 character encoding string. The double backslash (\\) is an escaped backslash, which tells the application firewall to interpret it as a literal backslash. If you included only one backslash, the application firewall would instead interpret the following left square bracket ([) as a literal character instead of the opening of a character class, which would break the expression.
- Allow users to access all GIF (.gif), JPEG (.jpg and .jpeg), and PNG (.png) format graphics at www.example.com:

  ^http://www[.]example[.]com/([0-9A-Za-z][0-9A-Za-z_-]*/)*
  [0-9A-Za-z][0-9A-Za-z_.-]*[.](gif|jpe?g|png)$
- Allow users to access CGI (.cgi) and PERL (.pl) scripts, but only in the CGI-BIN directory:

  ^http://www[.]example[.]com/CGI-BIN/[0-9A-Za-z][0-9A-Za-z_.-]*[.](cgi|pl)$
- Allow users to access Microsoft Office and other document files in the docsarchive directory:

  ^http://www[.]example[.]com/docsarchive/[0-9A-Za-z][0-9A-Za-z_-.]*[.](doc|xls|pdf|ppt)$

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions that you write. Make sure that they define exactly the URL you want to add as an exception, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results you do not want, such as blocking access to web content that you did not intend to block or allowing an attack that the Start URL check would otherwise have blocked.

Deny URL Check

Feb 03, 2014

The Deny URL check examines and blocks connections to URLs that are commonly accessed by hackers and malicious code. This check contains a list of URLs that are common targets of hackers or malicious code and that rarely if ever appear in legitimate requests. You can also add URLs or URL patterns to the list. The Deny URL check prevents attacks against various security weaknesses known to exist in web server software or on many web sites.

The Deny URL check takes priority over the Start URL check, and thus denies malicious connection attempts even when a Start URL relaxation would normally allow a request to proceed.

In the Modify Deny URL Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions.

If you use the command-line interface, you can enter the following command to configure the Deny URL Check:

- set appfw profile <name> -denyURLAction [**block**] [**log**] [**stats**] [**none**]

To create and configure your own deny URLs, you must use the configuration utility. On the Checks tab of the Modify Deny URL Check dialog box, click Add to open the Add Deny URL dialog box, or select an existing user-defined deny URL and click Open to open the Modify Deny URLdialog box. Either dialog box provides the same options for creating and configuring a deny URL.

Following are examples of Deny URL expressions:

- Do not allow users to access the image server at `images.example.com` directly:

  ```
  ^http://images[.]example[.]com$
  ```
- Do not allow users to access CGI (`.cgi`) or PERL (`.pl`) scripts directly:

  ```
  ^http://www[.]example[.]com/([0-9A-Za-z][0-9A-Za-z_-]*/)*
  [0-9A-Za-z][0-9A-Za-z_.-]*[.](cgi|pl)$
  ```
- Here is the same deny URL, modified to support non-ASCII characters:

  ```
  ^http://www[.]example[.]com/(([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])
  ([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*/)*([0-9A-Za-z]|\\x[0-9A-Fa-f][0-9A-Fa-f])
  ([0-9A-Za-z_-]|\\x[0-9A-Fa-f][0-9A-Fa-f])*[.](cgi|pl)$
  ```

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the URL or pattern that you want to block, and nothing else. Careless use of wildcards, and especially of the dot-asterisk (.*) metacharacter/wildcard combination, can have results that you do not want, such as blocking access to web content that you did not intend to block.

XML Protection Checks

Mar 28, 2012

The XML Protection checks examine requests for XML-based attacks of all types.

Caution: The XML security checks apply only to content that is sent with an HTTP content-type header of text/xml. If the content-type header is missing, or is set to a different value, all XML security checks are bypassed. If you plan to protect XML or Web 2.0 web applications, the webmasters of each web server that hosts those applications should ensure that the proper HTTP content-type header is sent.

XML Format Check

Feb 20, 2014

The XML Format check examines the XML format of incoming requests and blocks those requests that are not well formed or that do not meet the criteria in the XML specification for properly-formed XML documents. Some of those criteria are:

- An XML document must contain only properly-encoded Unicode characters that match the Unicode specification.
- No special XML syntax characters—such as < , > and &—can be included in the document except when used in XML markup.
- All begin, end, and empty-element tags must be correctly nested, with none missing or overlapping.
- XML element tags are case-sensitive. All beginning and end tags must match exactly.
- A single root element must contain all the other elements in the XML document.

A document that does not meet the criteria for well-formed XML does not meet the definition of an XML document. Strictly speaking, it is not XML. However, not all XML applications and web services enforce the XML well-formed standard, and not all handle poorly-formed or invalid XML correctly. Inappropriate handling of a poorly-formed XML document can cause security breaches. The purpose of the XML Format check is to prevent a malicious user from using a poorly-formed XML request to breach security on your XML application or web service.

If you use the wizard or the configuration utility, in the Modify XML Format Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions.

If you use the command-line interface, you can enter the following command to configure the XML Format Check:

- set appfw profile <name> -xmlFormatAction [**block**] [**log**] [**stats**] [**none**]

You cannot configure exceptions to the XML Format check. You can only enable or disable it.

XML Denial-of-Service Check

Nov 14, 2013

The XML Denial of Service (XML DoS or XDoS) check examines incoming XML requests to determine whether they match the characteristics of a denial-of-service (DoS) attack, and blocks those requests that do. The purpose of the XML DoS check is to prevent an attacker from using XML requests to launch a denial-of-service attack on your web server or web site.

If you use the wizard or the configuration utility, in the Modify XML Denial-of-Service Check dialog box, on the General tab you can enable or disable the Block, Log, Statistics, and Learn actions:

If you use the command-line interface, you can enter the following command to configure the XML Denial-of-Service check:

- set appfw profile <name> -xmlDoSAction [**block**] [**log**] [**learn**] [**stats**] [**none**]

To configure individual XML Denial-of-Service rules, you must use the configuration utility. On the Checks tab of the Modify XML Denial-of-Service Check dialog box, select a rule and click Open to open the Modify XML Denial-of-Service dialog box for that rule. The individual dialog boxes differ for the different rules but are extremely simple. Some only allow you to enable or disable the rule; others allow you to modify a number by typing a new value in a text box.

The individual XML Denial-of-Service rules are:

Maximum Element Depth

Restrict the maximum number of nested levels in each individual element to 256. If this rule is enabled, and the application firewall detects an XML request with an element that has more than the maximum number of allowed levels, it blocks the request. You can modify the maximum number of levels to any value from one (1) to 65,535.

Maximum Element Name Length

Restrict the maximum length of each element name to 128 characters. This includes the name within the expanded namespace, which includes the XML path and element name in the following format:

{http://prefix.example.com/path/}target_page.xml

The user can modify the maximum name length to any value between one (1) character and 65,535.

Maximum # Elements

Restrict the maximum number of any one type of element per XML document to 65,535. You can modify the maximum number of elements to any value between one (1) and 65,535.

Maximum # Element Children

Restrict the maximum number of children (including other elements, character information, and comments) each individual element is allowed to have to 65,535. You can modify the maximum number of element children to any value between one (1) and 65,535.

Maximum # Attributes

Restrict the maximum number of attributes each individual element is allowed to have to 256. You can modify the maximum number of attributes to any value between one (1) and 256.

Maximum Attribute Name Length

Restrict the maximum length of each attribute name to 128 characters. You can modify the maximum attribute name

length to any value between one (1) and 2,048.

Maximum Attribute Value Length

Restrict the maximum length of each attribute value to 2048 characters. You can modify the maximum attribute name length to any value between one (1) and 2,048.

Maximum Character Data Length

Restrict the maximum character data length for each element to 65,535. You can modify the length to any value between one (1) and 65,535.

Maximum File Size

Restrict the size of each file to 20 MB. You can modify the maximum file size to any value.

Minimum File Size

Require that each file be at least 9 bytes in length. You can modify the minimum file size to any positive integer representing a number of bytes.

Maximum # Entity Expansions

Limit the number of entity expansions allowed to the specified number. Default: 1024.

Maximum Entity Expansion Depth

Restrict the maximum number of nested entity expansions to no more than the specified number. Default: 32.

Maximum # Namespaces

Limit the number of namespace declarations in an XML document to no more than the specified number. Default: 16.

Maximum Namespace URI Length

Limit the URL length of each namespace declaration to no more than the specified number of characters. Default: 256.

Block Processing Instructions

Block any special processing instructions included in the request. This rule has no user-modifiable values.

Block DTD

Block any document type definitions (DTD) included with the request. This rule has no user-modifiable values.

Block External Entities

Block all references to external entities in the request. This rule has no user-modifiable values.

SOAP Array Check

Enable or disable the following SOAP array checks:

- **Maximum SOAP Array Size.** The maximum total size of all SOAP arrays in an XML request before the connection is blocked. You can modify this value. Default: 20000000.
- **Maximum SOAP Array Rank.** The maximum rank or dimensions of any single SOAP array in an XML request before the connection is blocked. You can modify this value. Default: 16.

XML Cross-Site Scripting Check

Jan 15, 2015

The XML Cross-Site Scripting check examines the user requests for possible cross-site scripting attacks in the XML payload. If it finds a possible cross-site scripting attack, it blocks the request.

To prevent misuse of the scripts on your protected web services to breach security on your web services, the XML Cross-Site Scripting check blocks scripts that violate the same origin rule, which states that scripts should not access or modify content on any server but the server on which they are located. Any script that violates the same origin rule is called a cross-site script, and the practice of using scripts to access or modify content on another server is called cross-site scripting. The reason cross-site scripting is a security issue is that a web server that allows cross-site scripting can be attacked with a script that is not on that web server, but on a different web server, such as one owned and controlled by the attacker.

Unfortunately, many companies have a large installed base of JavaScript-enhanced web content that violates the same origin rule. If you enable the XML Cross-Site Scripting check on such a site, you have to generate the appropriate exceptions so that the check does not block legitimate activity.

Actions: If you use the wizard or the configuration utility, in the XML Cross-Site Scripting Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions.
If you use the command-line interface, you can enter the following command to configure the actions for the XML Cross-Site Scripting Check:

- set appfw profile <name> -XMLXSSAction [**block** [**log**] [**stats**] [**none**]

Relaxations: You can use the configuration utility to specify relaxations for the XML Cross-Site Scripting check. On the Checks tab of the Modify XML Cross-Site Scripting Check dialog box, click Add to open the Add XML Cross-Site Scripting Check Relaxation dialog box, or select an existing relaxation and click Open to open the Modify XML Cross-Site Scripting Check Relaxation dialog box to edit an existing rule. Either dialog box provides the same options for configuring a relaxation, as described in "Manual Configuration By Using the Configuration Utility."

The XML Cross-Site Scripting check relaxation rules have the following parameters :

- **Name**: You can use literal strings or Regular Expressions to configure the name. The following expression exempts all elements beginning with the string name_ followed by a string of upper- and lower-case letters or numbers that is at least two characters long and no more than fifteen characters long:

  ```
  ^name_[0-9A-Za-z]{2,15}$
  ```

Caution: Regular expressions are powerful. Especially if you are not thoroughly familiar with PCRE-format regular expressions, double-check any regular expressions you write. Make sure that they define exactly the name that you want to add as an exception, and nothing else. Careless use of Regular Expressions can have results that you do not want, such as blocking access to web content that you did not intend to block or allowing an attack that the XML Cross-Site Scripting check would otherwise have blocked.

- **Location** You can specify the Location of the Cross-site Scripting Check exception in your XML payload. The option ELEMENT is selected by Default. You can change it to select ATTRIBUTE.

- **Comment** This is an optional field. You can use up to a 255 character long string to describe the purpose of this relaxation Rule.

XML SQL Injection Check

Dec 16, 2013

The XML SQL Injection check examines both the headers and the bodies of user requests for possible XML SQL Injection attacks. If it finds injected SQL, it blocks the request.

To prevent misusing the scripts on your protected web services to breach security on your web services, the XML SQL Injection check blocks scripts that violate the same origin rule, which states that scripts should not access or modify content on any server but the server on which they are located. Any script that violates the same origin rule is called a cross-site script, and the practice of using scripts to access or modify content on another server is called XML SQL Injection. The reason XML SQL Injection is a security issue is that a web server that allows XML SQL Injection can be attacked with a script that is not on that web server, but on a different web server, such as one owned and controlled by the attacker.

Unfortunately, many companies have a large installed base of JavaScript-enhanced web content that violates the same origin rule. If you enable the XML SQL Injection check on such a site, you have to generate the appropriate exceptions so that the check does not block legitimate activity. In addition, to prevent blocking of legitimate requests, this check ignores cookies that were set by the server, even if they contain elements that the Cookie Consistency check would otherwise block. You should keep this in mind when configuring this check.

Note: To prevent blocking of legitimate requests, this check ignores cookies that were set by the server, even if they contain elements that the Cookie Consistency check would otherwise block.
If you use the wizard or the configuration utility, in the Modify XML SQL Injection Check dialog box, on the General tab you can enable or disable Block, Log, and Statistics actions, and the following parameters:

- **Restrict checks to fields containing SQL special characters.** If you configure the application firewall to check only fields that contain SQL special strings, the application firewall skips web form fields that do not contain special characters. Since most SQL servers do not process SQL commands that are not preceded by a special character, enabling this parameter can significantly reduce the load on the application firewall and speed up processing without placing your protected web sites at risk.

- **SQL comments handling.** By default, the application firewall checks all SQL comments for injected SQL commands. Many SQL servers ignore anything in a comment, however, even if it is preceded by an SQL special character. For faster processing, if your SQL server ignores comments, you can configure the application firewall to skip comments when examining requests for injected SQL. The SQL comments handling options are:
  - **ANSI.** Skip ANSI-format SQL comments, which are normally used by UNIX-based SQL databases.
  - **Nested.** Skip nested SQL comments, which are normally used by Microsoft SQL Server.
  - **ANSI/Nested.** Skip comments that adhere to both the ANSI and nested SQL comment standards. Comments that match only the ANSI standard, or only the nested standard, are checked for injected SQL.
    Caution: In most cases, you should not choose the Nested or the ANSI/Nested option unless your database runs on Microsoft SQL Server. Most other types of SQL server software do not recognize nested comments. If nested comments appear in a request directed to another type of SQL server, they may indicate an attempt to breach security on that server.
  - **Check all Comments.** Check the entire request for injected SQL, without skipping anything. The default setting.
- **Check Request headers.** If Request header checking is enabled, the application firewall examines the headers of requests for XML SQL Injection attacks, instead of just URLs.

  Caution: If you enable both request header checking and transformation, any SQL special characters found in headers

are also transformed. The Accept, Accept-Charset, Accept-Encoding, Accept-Language, Expect, and User-Agent headers normally contain semicolons (;), so enabling both Request header checking and transformation simultaneously may cause errors.

If you use the command-line interface, you can enter the following commands to configure the XML SQL Injection Check:

- set appfw profile <name> -XMLSQLInjectionAction [**block**] [**learn**] [**log**] [**stats**] [**none**]
- set appfw profile <name> -XMLSQLInjectionOnlyCheckFieldsWithSQLChars (**ON** | **OFF**)]
- set appfw profile <name> -XMLSQLInjectionParseComments ([**checkall**] | [**ansi** | **nested**] | [**ansinested**])

You configure the exceptions to the XML SQL Injection check by opening the Modify XML SQL Injection Check dialog box, Checks tab. An exception can consist of either a literal string or a PCRE-format regular expression. For information about adding, modifying, removing, enabling, or disabling exceptions, see "Manual Configuration By Using the Configuration Utility."

Following are examples of XML SQL Injection check relaxations:

- **Name element or attribute.** The following expression exempts all elements beginning with the string name_ followed by a string of upper- and lower-case letters or numbers that is at least two characters long and no more than fifteen characters long:

  ```
  ^name_[0-9A-Za-z]{2,15}$
  ```
- **URL element or attribute.** The following expression exempts URLs with hostnames of web.example.com, with a path up to four levels deep followed by an optional file name and extension, but no HTML or query symbols :

  ```
  ^https?://web[.]example[.]com(/[^<>?]{1,30}){0,4}(/[^<>?]{1,30})*$
  ```
- **URL element or attribute (special characters).** The following expression exempts URLs with hostnames of web.türkçe-example.com, with the same path and file restrictions as above:

  ```
  ^https?://web[.]t\xC3\xBCrk\xC3\xA7e-example[.]com(/[^<>?]{1,30}){0,4}(/[^<>?]{1,30})*$
  ```

XML Attachment Check

Feb 20, 2014

The XML Attachment check examines incoming requests for malicious attachments, and it blocks those requests that contain attachments that might breach applications security. The purpose of the XML Attachment check is to prevent an attacker from using an XML attachment to breach security on your server.

If you use the wizard or the configuration utility, in the Modify XML Attachment Check dialog box, on the General tab you can enable or disable the Block, Learn, Log, Statistics, and Learn actions:

If you use the command-line interface, you can enter the following command to configure the XML Attachment Check:

- set appfw profile <name> -xmlAttachmentAction [**block**] [**learn**] [**log**] [**stats**] [**none**]

You must configure the other XML Attachment check settings in the configuration utility. In the Modify XML Attachment Check dialog box, on the Checks tab, you can configure the following settings:

- **Maximum Attachment Size.** Allow attachments that are no larger than the maximum attachment size you specify. To enable this option, first select the Enabled check box, and then type the maximum attachment size in bytes in the Size text box.
- **Attachment Content Type.** Allow attachments of the specified content type. To enable this option, first select the Enabled check box, and then enter a regular expression that matches the Content-Type attribute of the attachments that you want to allow.
  - You can type the URL expression directly in the text window. If you do so, you can use the Regex Tokens menu to enter a number of useful regular expressions at the cursor instead of typing them manually.
  - You can click Regex Editor to open the Add Regular Expression dialog box and use it to construct the URL expression.

Web Services Interoperability Check

Oct 30, 2013

The Web Services Interoperability (WS-I) check examines both requests and responses for adherence to the WS-I standard, and blocks those requests and responses that do not adhere to this standard. The purpose of the WS-I check is to block requests that might not interact with other XML appropriately. An attacker can use inconsistencies in interoperability to launch an attack on your XML application.

If you use the wizard or the configuration utility, in the Modify Web Services Interoperability Check dialog box, on the General tab you can enable or disable the Block, Log, Statistics, and Learn actions.

If you use the command-line interface, you can enter the following command to configure the Web Services Interoperability check:

- set appfw profile <name> -xmlWSIAction [**block**] [**log**] [**learn**] [**stats**] [**none**]

To configure individual Web Services Interoperability rules, you must use the configuration utility. On the Checks tab of the Modify Web Services Interoperability Check dialog box, select a rule and click Enable or Disable to enable or disable the rule. You can also click Open to open the Web Services Interoperability Detail message box for that rule. The message box displays read-only information about the rule. You cannot modify or make other configuration changes to any of these rules.

XML Message Validation Check

Feb 20, 2014

The XML Message Validation check examines requests that contain XML messages to ensure that they are valid. If a request contains an invalid XML message, the application firewall blocks the request. The purpose of the XML Validation check is to prevent an attacker from using specially constructed invalid XML messages to breach the security of your application.

If you use the wizard or the configuration utility, in the Modify XML Message Validation Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions.

If you use the command-line interface, you can enter the following command to configure the XML Message Validation Check:

- set appfw profile <name> -xmlValidationAction [**block**] [**log**] [**stats**] [**none**]

You must use the configuration utility to configure the other XML Validation check settings. In the Modify XML Message Validation Check dialog box, on the Checks tab, you can configure the following settings:

- **XML Message Validation.** Use one of the following options to validate the XML message:
  - **SOAP Envelope.** Validate only the SOAP envelope of XML messages.
  - **WSDL.** Validate XML messages by using an XML SOAP WSDL. If you choose WSDL validation, in the WSDL Object drop-down list you must choose a WSDL. If you want to validate against a WSDL that has not already been imported to the application firewall, you can click the Import button to open the Manage WSDL Imports dialog box and import your WSDL. See "WSDL" for more information.
    - If you want to validate the entire URL, leave the Absolute radio button in the End Point Check button array selected. If you want to validate only the portion of the URL after the host, select the Relative radio button.
    - If you want the application firewall to enforce the WSDL strictly, and not allow any additional XML headers not defined in the WSDL, you must clear the Allow additional headers not defined in the WSDL check box.
      Caution: If you uncheck the Allow Additional Headers not defined in the WSDL check box, and your WSDL does not define all XML headers that your protected XML application or Web 2.0 application expects or that a client sends, you may block legitimate access to your protected service.
  - **XML Schema.** Validate XML messages by using an XML schema. If you choose XML schema validation, in the XML Schema Object drop-down list you must choose an XML schema. If you want to validate against an XML schema that has not already been imported to the application firewall, you can click the Import button to open the Manage XML Schema Imports dialog box and import your WSDL. See "WSDL" for more information.
- **Response Validation.** By default, the application firewall does not attempt to validate responses. If you want to validate responses from your protected application or Web 2.0 site, select the Validate Response check box. When you do, the Reuse the XML Schema specified in request validation check box and the XML Schema Object drop-down list are activated.
  - Check the Reuse XML Schema check box to use the schema you specified for request validation to do response validation as well.
    Note: If you check this check box, the XML Schema Object drop-down list is grayed out.
  - If you want to use a different XML schema for response validation, use the XML Schema Object drop-down list to select or upload that XML schema .

XML SOAP Fault Filtering Check

Oct 30, 2013

The XML SOAP Fault Filtering check examines responses from your protected web services and filters out XML SOAP faults. This prevents leaking of sensitive information to attackers.

If you use the wizard or the configuration utility, in the Modify XML SOAP Fault Filtering Check dialog box, on the General tab you can enable or disable the Block, Log, and Statistics actions, and the Remove action, which removes SOAP faults before forwarding the response to the user.

If you use the command-line interface, you can enter the following command to configure the XML SOAP Fault Filtering Check:

set appfw profile <name> -XMLSOAPFaultAction [**block**] [**log**] [**stats**] [**none**]

You cannot configure exceptions to the XML SOAP Fault Filtering check. You can only enable or disable it.

Managing Content Types

Jun 12, 2014

Web servers usually add a Content-Type header that contains a MIME/type definition for the type of content in each file that the web server serves to users. Web servers serve many different types of content. For example, standard HTML is assigned the "text/html" MIME type. JPG images are assigned the "image/jpeg" or "image/jpg" content type. A normal web server can serve dozens or hundreds of different types of content, all defined in the Content Type header by an assigned MIME/type.

Many application firewall filtering rules are designed to filter specific types of content. Because filtering rules that apply to one type of content (such as HTML) are often inappropriate when filtering a different type of content (such as images), the application firewall attempts to determine the content type of requests and responses before it filters them. When a web server or browser does not add a Content-Type header to a request or response, the application firewall applies a default content type to the connection and filters the content accordingly.

The default content type is normally "application/octet-stream", the most generic MIME/type definition.This MIME/type is appropriate for any type of content that a web server is likely to serve, but also does not provide much information to the application firewall to allow it to choose appropriate filtering. If a protected web server on your network is configure to add accurate content type headers to the content it serves, or serves only one type of content, you can create a profile for that web server and assign a different default content type to it to improve both the speed and the accuracy of filtering.

You can also configure a list of allowed response content types for a specific profile. When this feature is configured, if the application firewall filters a response that does not match one of the allowed content types, it blocks the response.

Requests must always be of either the "application/x-www-form-urlencoded" or "multipart/form-data" types. The application firewall bypasses any request that has any other content type designated.

Note: You cannot include the "application/x-www-form-urlencoded" or "multipart/form-data" content types on the allowed response content types list.

To set the default request content type by using the command line interface

At the command prompt, type the following commands:

- set appfw profile <name> -requestContentType <type>
- save ns config

Example

The following example sets the "text/html" content type as the default for the specified profile:

```
set appfw profile profile1 -requestContentType "text/html"
save ns config
```

To remove the user-defined default request content type by using the command line interface

At the command prompt, type the following commands:

- unset appfw profile <name> -requestContentType <type>
- save ns config

Example

The following example unsets the default content type of "text/html" for the specified profile, allowing the type to revert to "application/octet-stream":

```
unset appfw profile profile1 -requestContentType "text/html"
save ns config
```

To set the default response content type by using the command line interface

At the command prompt, type the following commands:

- set appfw profile <name> -responseContentType <type>
- save ns config

Example

The following example sets the "text/html" content type as the default for the specified profile:

```
set appfw profile profile1 -responseContentType "text/html"
save ns config
```

To remove the user-defined default response content type by using the command line interface

At the command prompt, type the following commands:

- unset appfw profile <name> -responseContentType <type>
- save ns config

Example

The following example unsets the default content type of "text/html" for the specified profile, allowing the type to revert to "application/octet-stream":

```
unset appfw profile profile1 -responseContentType "text/html"
save ns config
```

To add a content type to the allowed content types list by using the command line interface

At the command prompt, type the following commands:

- bind appfw profile <name> -ContentType <contentTypeName>
- save ns config

Example

The following example adds the "text/shtml" content type to the allowed content types list for the specified profile:

```
bind appfw profile profile1 -contentType "text/shtml"
save ns config
```

To remove a content type from the allowed content types list by using the command line interface

At the command prompt, type the following commands:

- unbind appfw profile <name> -ContentType <contentTypeName>

- save ns config

Example

The following example removes the "text/shtml" content type from the allowed content types list for the specified profile:

```
unbind appfw profile profile1 -contentType "text/shtml"
save ns config
```

To manage the default and allowed content types by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, select the profile that you want to configure, and then click Open. The Configure Application Firewall Profile dialog box is displayed.
3. In the Configure Application Firewall Profile dialog box, click the Settings tab
4. On the Settings tab, scroll down about halfway to the Content Type area.
5. In the Content Type area, configure the default request or response content type:
   - To configure the default request content type, type the MIME/type definition of the content type you want to use in the Default Request text box.
   - To configure the default response content type, type the MIME/type definition of the content type you want to use in the Default Response text box.
   - To create a new allowed content type, click Add. The Add Allowed Content Type dialog box is displayed.
   - To edit an existing allowed content type, select that content type, and then click Open. The Modify Allowed Content Type dialog box is displayed.
6. To manage the allowed content types, click Manage Allowed Content Types.
7. To add a new content type or modify an existing content type, click Add or Open, and in the Add Allowed Content Type or Modify Allowed Content Type dialog box, do the following steps.
   1. Select/clear the Enabled check box to include the content type in, or exclude it from, the list of allowed content types.
   2. In the Content Type text box, type a regular expression that describes the content type that you want to add, or change the existing content type regular expression.
      Content types are formatted exactly as MIME type descriptions are.

      Note: You can include any valid MIME type on the allowed contents type list. Since many types of document can contain active content and therefore could potentially contain malicious content, you should exercise caution when adding MIME types to this list.
   3. In the Comments text box, add an optional comment that describes the reason for adding this particular MIME type to the allowed contents type list.
   4. Click Create or OK to save your changes.
8. Click Close to close the Manage Allowed Content Types dialog box and return to the Settings tab.
9. Click OK to save your changes.

Profiles

Feb 03, 2014

A profile is a collection of security settings that are used to protect specific types of web content or specific parts of your web site. In a profile, you determine how the application firewall applies each of its filters (or checks) to requests to your web sites, and responses from them. The application firewall supports two types of profile: four built-in (default) profiles that do not require further configuration, and user-defined profiles that do require further configuration.

Built-In Profiles

The four application firewall built-in profiles provide simple protection for applications and web sites that either do not require protection, or that should not be directly accessed by users at all. These profile types are:

- **APPFW_BYPASS.** Skips all application firewall filtering and sends the unmodified traffic to the protected application or web site, or to the client.
- **APPFW_RESET.** Resets the connection, requiring that the client re-establish his or her session by visiting a designated start page.
- **APPFW_DROP.** Drops all traffic to or from the protected application or web site, and sends no response of any kind to the client.
- **APPFW_BLOCK.** Blocks traffic to or from the protected application or web site.

You use the built-in profiles exactly as you do user-defined profiles, by configuring a policy that selects the traffic to which you want to apply the profile and then associating the profile with your policy. Since you do not have to configure a built-in policy, it provides a quick way to allow or block specified types of traffic or traffic that is sent to specific applications or web sites.

User-Defined Profiles

User-defined profiles are profiles that are build and configured by users. Unlike the default profiles, you must configure a user-defined profile before it will be of use filtering traffic to and from your protected applications.

There are three types of user-defined profile:

- **HTML.** Protects HTML-based web pages.
- **XML.** Protects XML-based web services and web sites.
- **Web 2.0.** Protects Web 2.0 content that combines HTML and XML content, such as ATOM feeds, blogs, and RSS feeds.

The application firewall has a number of security checks, all of which can be enabled or disabled, and configured in a number of ways in each profile. Each profile also has a number of settings that control how it handles different types of content. Finally, rather than manually configuring all of the security checks, you can enable and configure the learning feature. This feature observes normal traffic to your protected web sites for a period of time, and uses those observations to provide you with a tailored list of recommended exceptions (*relaxations*) to some security checks, and additional rules for other security checks.

During initial configuration, whether by using the Application Firewall Wizard or manually, you normally create one general purpose profile to protect all content on your web sites that is not covered by a more specific profile. After that, you can create as many specific profiles as you want to protect more specialized content.

The Profiles pane consists of a table that contains the following elements:

Name. Displays all the application firewall profiles configured in the appliance.

Bound signature. Displays the signatures object that is bound to the profile in the previous column, if any.

Policies. Displays the application firewall policy that invokes the profile in the leftmost column of that row, if any.

Comments. Displays the comment associated with the profile in the leftmost column of that row, if any.

Profile Type. Displays the type of profile. Types are Built-In, HTML, XML, and Web 2.0.

Above the table is a row of buttons and a drop-down list that allow you to create, configure, delete, and view information about your profiles:

- **Add.** Add a new profile to the list.
- **Edit.** Edit the selected profile.
- **Delete.** Delete the selected profile from the list.
- **Statistics.** View the statistics for the selected profile.
- **Action.** Drop-down list that contains additional commands. Currently allows you to import a profile that was exported from another application firewall configuration.

Creating Application Firewall Profiles

Jun 12, 2014

You can create an application firewall profile in one of two ways: by using the command line, and by using the configuration utility. Creating a profile by using the command line requires that you specify options on the command line. The process is similar to that of configuring an existing profile, and with a few exceptions the two commands take the same parameters.

Creating a profile by using the configuration utility requires that you specify only two options. You specify basic or advanced *defaults*, the default configuration for the various security checks and settings that are part of a profile, and choose the profile *type* to match the type of content that the profile is intended to protect. You can also, optionally, add a comment. After you create the profile, you must then configure it by selecting it in the data pane, and then clicking Edit.

If you plan to use the learning feature or to enable and configure a large number of advanced protections, you should choose advanced defaults. In particular, if you plan to configure either of the SQL injection checks, either of the cross-site scripting checks, any check that provides protection against Web form attacks, or the cookie consistency check, you should plan to use the learning feature. Unless you include the proper exceptions for your protected Web sites when configuring these checks, they can block legitimate traffic. Anticipating all of the necessary exceptions without creating any that are too broad is difficult. The learning feature makes this task much easier. Otherwise, basic defaults are quick and should provide the protection that your web applications need.

There are three profile types:

- **HTML.** Protects standard HTML-based web sites.
- **XML.** Protects XML-based web services and web sites.
- **Web 2.0 (HTML XML).** Protects sites that contain both HTML and XML elements, such as ATOM feeds, blogs, and RSS feeds.

There are also a few restrictions on the name that you can give to a profile. A profile name cannot be the same as the name assigned to any other profile or action in any feature on the NetScaler appliance. Certain action or profile names are assigned to built-in actions or profiles, and can never be used for user profiles. A complete list of disallowed names can be found in the Application Firewall ProfileSupplemental Information. If you attempt to create a profile with a name that has already been used for an action or a profile, an error message is displayed and the profile is not created.

To create an application firewall profile by using the command line interface

At the command prompt, type the following commands:

- add appfw profile <name> [-defaults (**basic** | **advanced**)]
- set appfw profile <name> -type (**HTML** | **XML** | **HTML XML**)
- set appfw profile <name> -comment "<comment>"
- save ns config

Example

The following example adds a profile named pr-basic, with basic defaults, and assigns a profile type of HTML. This is the appropriate initial configuration for a profile to protect an HTML Web site.

```
add appfw profile pr-basic -defaults basic -comment "Simple profile for web sites."
set appfw profile pr-basic -type HTML
```

save ns config

To create an application firewall profile by using the configuration utility

Creating an application firewall profile requires that you specify only a few configuration details.

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, click Add.
3. In the Create Application Firewall Profile dialog box, type a name for your profile.
   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) symbols.
4. Choose the profile type from the drop-down list.
5. Click Create, and then click Close.

Configuring Application Firewall Profiles

Jun 12, 2014

To configure a user-defined application firewall profile, first configure the security checks, which are called *deep protections* or *advanced protections* in the application firewall wizard. Certain checks require configuration if you are to use them at all. Others have default configurations that are safe but limited in scope; your web sites might need or benefit from a different configuration that takes advantage of additional features of certain security checks.

After you have configured the security checks, you can also configure a number of other settings that control the behavior, not of a single security check, but the application firewall feature. The default configuration is sufficient to protect most web sites, but you should review them to make sure that they are right for your protected web sites.

For more information about the application firewall security checks, see "Advanced Protections."

To configure an application firewall profile by using the command line

At the command prompt, type the following commands:

- set appfw profile <name> <arg1> [<arg2> ...]
  where:
  - <arg1> = a parameter and any associated options.
  - <arg2> = a second parameter and any associated options.
  - ... = additional parameters and options.

  For descriptions of the parameters to use when configuring specific security checks, see "Advanced Protections."
- save ns config

Example

The following example shows how to enable blocking for the HTML SQL Injection and HTML Cross-Site Scripting checks in a profile named pr-basic. This command enables blocking for those actions while making no other changes to the profile.

```
set appfw profile pr-basic -crossSiteScriptingAction block
        -SQLInjectionAction block
```

To configure an application firewall profile by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, select the profile that you want to configure, and then click Open.
3. In the Configure Application Firewall Profile dialog box, on the Security Checks tab, configure the security checks.
   - To enable or disable an action for a check, in the list, select or clear the check box for that action.
   - To configure other parameters for those checks that have them, in the list, click the blue chevron to the far right of that check. In the dialog box that appears, configure the parameters. These vary from check to check.
     You can also select a check and, at the bottom of the dialog box, click Open to display the Configure Relaxation dialog box or Configure Rule dialog box for that check. These dialog boxes also vary from check to check. Most of them include a Checks tab and a General tab. If the check supports relaxations or user-defined rules, the Checks tab includes an Add button, which opens yet another dialog box, in which you can specify a relaxation or rule for the check. (A relaxation is a rule for exempting specified traffic from the check.) If relaxations have already been configured, you can select one and click Open to modify it.

- To review learned exceptions or rules for a check, select the check, and then click Learned Violations. In the Manage Learned Rules dialog box, select each learned exception or rule in turn.
  - To edit the exception or rule, and then add it to the list, click Edit & Deploy.
  - To accept the exception or rule without modification, click Deploy.
  - To remove the exception or rule from the list, click Skip.
- To refresh the list of exceptions or rules to be reviewed, click Refresh.
- To open the Learning Visualizer and use it to review learned rules, click Visualizer.
- To review the log entries for connections that matched a check, select the check, and then click Logs. You can use this information to determine which checks are matching attacks, so that you can enable blocking for those checks. You can also use this information to determine which checks are matching legitimate traffic, so that you can configure an appropriate exemption to allow those legitimate connections. For more information about the logs, see "Logs, Statistics, and Reports."
- To completely disable a check, in the list, clear all of the check boxes to the right of that check.

4. On the Settings tab, configure the profile settings.
   - To associate the profile with the set of signatures that you previously created and configured, under Common Settings, choose that set of signatures in the Signatures drop-down list.
     Note: You may need to use the scroll bar on the right of the dialog box to scroll down to display the Common Settings section.
   - To configure an HTML or XML Error Object, select the object from the appropriate drop-down list.
     Note: You must first upload the error object that you want to use in the Imports pane. For more information about importing error objects, see "Imports."
   - To configure the default XML Content Type, type the content type string directly into the Default Request and Default Response text boxes, or click Manage Allowed Content Types to manage the list of allowed content types. ">>More...."
5. If you want to use the learning feature, click Learning, and configure the learning settings for the profile, as described in "Configuring and Using the Learning Feature".
6. Click OK to save your changes and return to the Profiles pane.

Changing an Application Firewall Profile Type

Jun 12, 2014

If you chose the wrong profile type for an application firewall profile, or the type of content on the protected web site has changed, you can change the profile type.

Note: When you change the profile type, you lose all configuration settings and learned relaxations or rules for the features that the new profile type does not support. For example, if you change the profile type from Web 2.0 to XML, you lose any configuration options for Start URL, Form Field Consistency Check, and the other HTML-specific security checks. The configuration for any options that is supported by both the old and the new profile types remains unchanged.

To change an application firewall profile type by using the command line interface

At the command prompt, type the following commands:

- set appfw profile <name> -type (**HTML** | **XML** | **HTML XML**)
- save ns config

Example

The following example changes the type of a profile named pr-basic, from HTML to HTML XML, which is equivalent to the Web 2.0 type in the configuration utility.

```
set appfw profile pr-basic -type HTML XML
save ns config
```

To change an application firewall profile type by using the configuration utility.

1. Navigate to Security > Application Firewall > Policies.
2. In the details pane, click Change Profile Type.
3. In the Change Application Firewall Profile Type dialog box, Profile Type drop-down list, select a new profile type.
4. Click OK to save your changes and return to the Profiles pane.

Exporting and Importing an Application Firewall Profile

Oct 07, 2014

You can export application firewall profiles to your local computer as files, and import previously exported profile files. You might want to configure an application firewall in a test bed configuration and then export the profile or profiles so that you can import the profile configuration to your production NetScaler ADCs. You might also want to export a profile to back up your configuration before making changes so that you can easily roll the configuration back to a know state if necessary.

To export an application firewall profile by using the command line interface

At the command prompt, type the following commands:

- archive appfw profile <name> <archiveName>
  Make the following substitutions:
  - name—Name of the profile to archive.
  - archiveName—Name of the archive file to create.
- export appfw archive <archiveName> <target>
  Make the following substitutions:
  - archiveName—Name of the archive to export. (The same name as in the previous step.)
  - target—Full path and filename of the exported file on your local computer. Enclose in straight double quotation marks if the path or filename contains spaces.

Example

Assuming that your local computer uses the Windows 7 operating system, and you are logged onto your local computer as "jsmith", the following example exports a profile named pr-basic to the home directory.

```
archive appfw profile pr-basic pr-basic.tgz
export appfw pr-basic.tgz "C:\Users\jsmith\Documents\pr-basic.tgz"
```

To export an application firewall profile by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, select a profile to export, and then click Export.
3. Choose the local path and filename for the exported file.
   - You can accept the default choice, which consists of the path to your home directory or folder and a filename of the profile name plus the extension .tgz, which indicates a Unix-style tar archive that is compressed by gzip.
   - You can type a new path and/or file name. The path must exist, and the filename must be a valid filename in your local computer's operating system. If you do not specify the .tgz extension, it is added automatically.
   - You can use the Browse dialog to locate the path and save the file under the default filename. (Recommended)
4. Click Export. The profile is exported and saved to your computer under the path and file name that you designated.

To import an application firewall profile by using the command line interface

At the command prompt, type the following command:

- import appfw archive <source> <archiveName>
  Make the following substitutions:
  - source—Full path and filename of the archive file to be imported from your local computer. Enclose in straight double quotation marks if the path or filename contains spaces.
  - archiveName—Name of the archive file on the NetScaler ADC.
- restore appfw profile <archiveName>

Example

Assuming that your local computer uses the Windows 7 operating system, and you are logged onto your local computer as "jsmith", the following example imports a profile named **pr-basic.tgz**, located in your home directory, to the application firewall and installs it as a profile named pr-basic.

```
import appfw archive "C:\Users\jsmith\Documents\pr-basic.tgz" pr-basic.tgz
restore appfw profile pr-basic.tgz
```

To import an application firewall profile by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the details pane, click Import. If you select a profile before you click Actions, the Import button does not appear. Select a different pane (such as Application Firewall > Policies) and then return to Application Firewall > Profiles to deselect the profile.
3. Choose the import type.
   - To import from a URL, accept the default selection, Import from URL.
   - To import a local file, select Import from Local File.
4. Specify the location of the profile to be imported.
   - You can type a URL, or a path and/or file name for the profile to be imported. The URL or path and filename must exist.
   - If you are importing a local file, you can use the Browse dialog to locate the path and filename of the profile to be imported. (Recommended)
5. Click Import. The profile is imported and appears in the Profiles pane.

Configuring and Using the Learning Feature

Jun 12, 2014

The learning feature is a repetitive pattern filter that observes activity on a web site or application protected by the application firewall, to determine what constitutes normal activity on that web site or application. It then generates a list of up to 2,000 suggested rules or exceptions (relaxations) for each security checks that includes support for the learning feature. Users normally find it easier to configure relaxations by using the learning feature than by entering the necessary relaxations manually.

The security checks that support the learning feature are:

- Start URL check
- Cookie Consistency check
- Form Field Consistency check
- Field Formats check
- CSRF Form Tagging check
- HTML SQL Injection check
- HTML Cross-Site Scripting check
- XML Denial-of-Service check
- XML Attachment check
- Web Services Interoperability check

You perform two different types of activities when using the learning feature. First, you enable and configure the feature to use it. You can use learning on all traffic to your protected web applications, or you can configure a list of IPs (called the *Add Trusted Learning Clients* list) from which the learning feature should generate recommendations. Second, after the feature has been enabled and has processed a certain amount of traffic to your protected web sites, you review the list of suggested rules and relaxations (learned rules) and mark each with one of the following designations:

- **Edit & Deploy.** The rule is pulled into the Edit dialog box so that you can modify it, and the modified form is deployed.
- **Deploy.** The unmodified learned rule is placed on the list of rules or relaxations for this security check.
- **Skip.** The learned rule is placed on a list of rules or relaxations that are not deployed, and that should not be learned again.

Although you can use the command line interface for basic configuration of the learning feature, the feature is designed primarily for configuration through the Application Firewall wizard or the configuration utility. You can perform only limited configuration of the learning feature by using the command line.

The wizard integrates configuration of learning features with configuration of the application firewall as a whole, and is therefore the easiest method for configuring this feature on a new NetScaler appliance or when managing a simple application firewall configuration. The configuration utility visualizer and manual interface both provide direct access to all learned rules for all security checks, and are therefore often preferable when you must review learned rules for a large number of security checks.

The learning database is limited to 20 MB in size, which is reached after approximately 2,000 learned rules or relaxations are generated per security check for which learning is enabled. If you do not regularly review and either approve or ignore learned rules and this limit is reached, an error is logged to the NetScaler log and no more learned rules are generated until you review the existing learned rules and relaxations.

If learning stops because the database has reached its size limit, you can restart learning either by reviewing the existing learned rules and relaxations or by resetting the learning data. After learned rules or relaxations are approved or ignored, they are removed from the database. After you reset the learning data, all existing learning data is removed from the database and it is reset to its minimum size. When the database falls below 20 MB in size, learning restarts automatically.

To configure the learning settings by using the command line interface

Specify the application firewall profile to be configured and, for each security check that you want to include in that profile, specify the minimum threshold or the percent threshold. The minimum threshold is an integer representing the minimum number of user sessions that the application firewall must process before it learns a rule or relaxation (default: 1). The percent threshold is an integer representing the percentage of user sessions in which the application firewall must observe a particular pattern (URL, cookie, field, attachment, or rule violation) before it learns a rule or relaxation (default: 0). Use the following commands:

- set appfw learningsettings <profileName> [-startURLMinThreshold <positive_integer>] [-startURLPercentThreshold <positive_integer>] [-cookieConsistencyMinThreshold <positive_integer>] [-cookieConsistencyPercentThreshold <positive_integer>] [-CSRFtagMinThreshold <positive_integer>] [-CSRFtagPercentThreshold <positive_integer>] [-fieldConsistencyMinThreshold <positive_integer>] [-fieldConsistencyPercentThreshold <positive_integer>] [-crossSiteScriptingMinThreshold <positive_integer>] [-crossSiteScriptingPercentThreshold <positive_integer>] [-SQLInjectionMinThreshold <positive_integer>] [-SQLInjectionPercentThreshold <positive_integer>] [-fieldFormatMinThreshold <positive_integer>] [-fieldFormatPercentThreshold <positive_integer>] [-XMLWSIMinThreshold <positive_integer>] [-XMLWSIPercentThreshold <positive_integer>] [-XMLAttachmentMinThreshold <positive_integer>] [-XMLAttachmentPercentThreshold <positive_integer>]
- save ns config

Example

The following example enables and configures the learning settings in the profile pr-basic for the HTML SQL Injection security check. This is an appropriate initial test bed learning configuration, where you have complete control over the traffic that is sent to the application firewall.

```
set appfw learningsettings pr-basic -SQLInjectionMinThreshold 10
set appfw learningsettings pr-basic -SQLInjectionPercentThreshold 70
save ns config
```

To reset learning settings to their defaults by using the command line interface

To remove any custom configuration of the learning settings for the specified profile and security check, and return the learning settings to their defaults, at the command prompt type the following commands:

- unset appfw learningsettings <profileName> [-startURLMinThreshold] [-startURLPercentThreshold] [-cookieConsistencyMinThreshold] [-cookieConsistencyPercentThreshold] [-CSRFtagMinThreshold] [-CSRFtagPercentThreshold] [-fieldConsistencyMinThreshold] [-fieldConsistencyPercentThreshold] [-crossSiteScriptingMinThreshold] [-crossSiteScriptingPercentThreshold] [-SQLInjectionMinThreshold] [-SQLInjectionPercentThreshold] [-fieldFormatMinThreshold] [-fieldFormatPercentThreshold] [-XMLWSIMinThreshold] [-XMLWSIPercentThreshold] [-XMLAttachmentMinThreshold] [-XMLAttachmentPercentThreshold]
- save ns config

To display the learning settings for a profile by using the command line interface

At the command prompt, type the following command:

show appfw learningsettings <profileName>

To display unreviewed learned rules or relaxations for a profile by using the command line interface

At the command prompt, type the following command:

show appfw learningdata <profileName> <securityCheck>

To remove specific unreviewed learned rules or relaxations from the learning database by using the command line interface

At the command prompt, type the following command:

rm appfw learningdata <profileName> (-startURL <expression> | -cookieConsistency <string> | (-fieldConsistency <string> <formActionURL>) | (-crossSiteScripting <string> <formActionURL>) | (-SQLInjection <string> <formActionURL>) | (-fieldFormat <string><formActionURL>) | (-CSRFTag <expression> <CSRFFormOriginURL>) | -XMLDoSCheck <expression> | -XMLWSICheck <expression> | -XMLAttachmentCheck <expression>) [-TotalXMLRequests]

Example

The following example removes all unreviewed learned relaxations for the pr-basic profile, HTML SQL Injection security check, that apply to the LastName form field.

rm appfw learningdata pr-basic -SQLInjection LastName

To remove all unreviewed learned data by using the command line interface

At the command prompt, type the following command:

reset appfw learningdata

To export learning data by using the command line interface

At the command prompt, type the following command:

export appfw learningdata <profileName> <securitycheck>[-target <string>]

Example

The following example exports learned relaxations for the pr-basic profile and the HTML SQL Injection security check to a comma-separated values (CSV) format file in the /var/learnt_data/ directory under the filename specified in the -target parameter.

export appfw learningdata pr-basic SQLInjection -target sqli_ld

To configure the Learning feature by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. In the Profiles pane, select the profile, and then click Open.
3. Click the Learning tab. At the top of the Learning tab is list of the security checks that are available in the current profile and that support the learning feature.
4. To configure the learning thresholds, select a security check, and then type the appropriate values in the following text boxes:
   - **Minimum number threshold.** Depending on which security check's learning settings you are configuring, the minimum number threshold might refer to the minimum number of total user sessions that must be observed, the minimum number of requests that must be observed, or the minimum number of times a specific form field must be observed, before a learned relaxation is generated. Default: 1

- **Percentage of times threshold.** Depending on which security check's learning settings you are configuring, the percentage of times threshold might refer to the percentage of total observed user sessions that violated the security check, the percentage of requests, or the percentage of times a form field matched a particular field type, before a learned relaxation is generated. Default: 0

5. To remove all learned data and reset the learning feature, so that it must start its observations again from the beginning, click Remove All Learned Data.
   Note: This button removes only learned recommendations that have not been reviewed and either approved or skipped. It does not remove learned relaxations that have been accepted and deployed.
6. To restrict the learning engine to traffic from a specific set of IPs, click Trusted Learning Clients, and add the IP addresses that you want to use to the list.
   1. To add an IP address or IP address range to the Trusted Learning Clients list, click Add.
   2. In the Add Trusted Learning Clients dialog box, Trusted Clients IP list box, type the IP address or an IP address range in CIDR format.
   3. In the Comments text area, type a comment that describes this IP address or range.
   4. Click Create to add your new IP address or range to the list.
   5. To modify an existing IP address or range, click the IP address or range, and then click Open. Except for the name, the dialog box that appears is identical to the Add Trusted Learning Clients dialog box.
   6. To disable or enable an IP address or range, but leave it on the list, click the IP address or range, and then click Disable or Enable, as appropriate.
   7. To remove an IP address or range completely, click the IP address or range, and then click Remove.
7. Click Close to return to the Configure Application Firewall Profile dialog box.
8. Click Close to close the Configure Application Firewall Profile dialog box, and return to the Application Firewall Profile screen.

To review learned rules or relaxations by using the configuration utility

1. Navigate to Security > Application Firewall > Profiles.
2. Select the security check for which you want to review learned rules or relaxations, and then click Manage Rules.
3. In the Manage Learned Rules dialog box, choose how you want to review the learned rules.
   - To review the actual learned patterns as displayed in the window, do nothing and proceed to the next step.
   - To review the learned data hierarchically as a branching tree, enabling you to choose general patterns that match many of the learned patterns, click Visualizer.
4. If you have chosen to review actual learned patterns, perform the following steps.
   1. Select the first learned relaxation and choose how to handle it.
      - To modify and then accept the relaxation, click Edit & Deploy, edit the relaxation regular expression, and then click OK.
      - To accept the relaxation without modifications, click Deploy.
      - To remove the relaxation from the list without deploying it, click Skip.
   2. Repeat the previous step to review each additional learned relaxation.
5. If you have chosen to use the Learning Visualizer, perform the following steps.
   1. In the branching hierarchical display, select a node that contains a learned pattern, and choose how to handle it. The screen area beneath the tree structure, under Regex of Selected Node, displays a generalized expression that matches all of the patterns in that node. If you want to display an expression that matches just one of the branches or just one of the leaves, select that branch or leaf.

- To modify and then accept the learned relaxation, click Edit & Deploy, edit the relaxation regular expression, and then click OK.
- To accept the relaxation without modifications, click Deploy.
- To remove the modification from the list without deploying it, click Skip.

2. Repeat the previous step to review other portions of the display.
3. Click Close to return to the Manage Learned Rules dialog box.

6. Click Close to return to the Configure Application Firewall Profile dialog box.
7. Click Close to close the Configure Application Firewall Profile dialog box, and return to the Application Firewall Profile screen.

Supplemental Information about Profiles

Oct 01, 2013

Following is supplemental information about particular aspects of application firewall profiles. This information explains how to include special characters in a security check rule or relaxation, and how to use variables when configuring profiles.

Configuration Variable Support

Instead of using static values, to configure the application firewall's security checks and settings, you can now use standard NetScaler named variables. By creating variables, you can more easily export and then import configurations to new NetScaler appliances, or update existing NetScaler appliances from a single set of configuration files. This simplifies updates when you use a test bed setup to develop a complex application firewall configuration that is tuned for your local network and servers and then transfer that configuration to your production NetScaler appliances.

You create application firewall configuration variables in the same manner as you do any other NetScaler named variables, following standard NetScaler conventions. To create a named expression variable by using the configuration utility, you use the "Add Expression dialog box." To create a named expression variable by using the NetScaler command line, you use the `add expression` command followed by the appropriate parameter.

The following URLs and expressions can be configured with variables instead of static values:

- **Start URL** (-starturl)
- **Deny URL** (-denyurl)
- **Form Action URL** for *Form Field Consistency Check* (-fieldconsistency)
- **Action URL** for *XML SQL Injection Check* (-xmlSQLInjection)
- **Action URL** for X*ML Cross-Site Scripting Check* (-xmlXSS)
- **Form Action URL** for *HTML SQL Injection Check* (-sqlInjection)
- **Form Action URL** for *Field Format Check* (-fieldFormat)
- **Form Origin URL** and **Form Action URL** for *Cross-Site Request Forgery (CSRF) Check* (-csrfTag)
- **Form Action URL** for *HTML Cross-Site Scripting Check* (-crossSiteScripting)
- **Safe Object** (-safeObject)
- **Action URL** for *XML Denial-of-Service (XDoS) check* (-XMLDoS)
- **URL** for *Web Services Interoperability check* (-XMLWSIURL)
- **URL** for *XML Validation check* (-XMLValidationURL)
- **URL** for *XML Attachment check* (-XMLAttachmentURL)

For more information, see "Policies and Expressions."

To use a variable in the configuration, you enclose the variable name between two at (@) symbols and then use it exactly as you would the static value that it replaces. For example, if you are configuring the Deny URL check by using the configuration utility and want to add the named expression variable myDenyURL to the configuration, you would type `@myDenyURL@` into the Add Deny URL dialog box, Deny URL text area. To do the same task by using the NetScaler command line, you would type `add appfw profile <name> -denyURLAction @myDenyURL@`.

PCRE Character Encoding Format

The NetScaler operating system supports direct entry of characters in the printable ASCII character set only—characters with hexadecimal codes between HEX 20 (ASCII 32) and HEX 7E (ASCII 127). To include a character with a code outside that range in your application firewall configuration, you must enter its UTF-8 hexadecimal code as a PCRE regular

expression.

A number of character types require encoding using a PCRE regular expression if you include them in your application firewall configuration as a URL, form field name, or Safe Object expression. They include:

- **Upper-ASCII characters.** Characters with encodings from HEX 7F (ASCII 128) to HEX FF (ASCII 255). Depending on the character map used, these encodings can refer to control codes, ASCII characters with accents or other modifications, non-Latin alphabet characters, and symbols not included in the basic ASCII set. These characters can appear in URLs, form field names, and safe object expressions.
- **Double-Byte characters.** Characters with encodings that use two 8-byte words. Double-byte characters are used primarily for representing Chinese, Japanese, and Korean text in electronic format. These characters can appear in URLs, form field names, and safe object expressions.
- **ASCII control characters.** Non-printable characters used to send commands to a printer. All ASCII characters with hexadecimal codes less than HEX 20 (ASCII 32) fall into this category. These characters should never appear in a URL or form field name, however, and would rarely if ever appear in a safe object expression.

The NetScaler appliance does not support the entire UTF-8 character set, but only the characters found in the following eight charsets:

- **English US (ISO-8859-1).** Although the label reads, "English US," the application firewall supports all characters in the ISO-8859-1 character set, also called the Latin-1 character set. This character set fully represents most modern western European languages and represents all but a few uncommon characters in the rest.
- **Chinese Traditional (Big5).** The application firewall supports all characters in the BIG5 character set, which includes all of the Traditional Chinese characters (ideographs) commonly used in modern Chinese as spoken and written in Hong Kong, Macau, Taiwan, and by many people of Chinese ethnic heritage who live outside of mainland China.
- **Chinese Simplified (GB2312).** The application firewall supports all characters in the GB2312 character set, which includes all of the Simplified Chinese characters (ideographs) commonly used in modern Chinese as spoken and written in mainland China.
- **Japanese (SJIS).** The application firewall supports all characters in the Shift-JIS (SJIS) character set, which includes most characters (ideographs) commonly used in modern Japanese.
- **Japanese (EUC-JP).** The application firewall supports all characters in the EUC-JP character set, which includes all characters (ideographs) commonly used in modern Japanese.
- **Korean (EUC-KR).** The application firewall supports all characters in the EUC-KR character set, which includes all characters (ideographs) commonly used in modern Korean.
- **Turkish (ISO-8859-9).** The application firewall supports all characters in the ISO-8859-9 character set, which includes all letters used in modern Turkish.
- **Unicode (UTF-8).** The application firewall supports certain additional characters in the UTF-8 character set, including those used in modern Russian.

When configuring the application firewall, you enter all non-ASCII characters as PCRE-format regular expressions using the hexadecimal code assigned to that character in the UTF-8 specification. Symbols and characters within the normal ASCII character set, which are assigned single, two-digit codes in that character set, are assigned the same codes in the UTF-8

character set. For example, the exclamation point (!), which is assigned hex code 21 in the ASCII character set, is also hex 21 in the UTF-8 character set. Symbols and characters from another supported character set have a paired set of hexadecimal codes assigned to them in the UTF-8 character set. For example, the letter a with an acute accent (á) is assigned UTF-8 code C3 A1.

The syntax you use to represent these UTF-8 codes in the application firewall configuration is "\xNN" for ASCII characters; "\xNN\xNN" for non-ASCII characters used in English, Russian, and Turkish; and "\xNN\xNN\xNN" for characters used in Chinese, Japanese, and Korean. For example, if you want to represent a ! in an application firewall regular expression as a UTF-8 character, you would type \x21. If you want to include an á, you would type \xC3\xA1.

Note: Normally you do not need to represent ASCII characters in UTF-8 format, but when those characters might confuse a web browser or an underlying operating system, you can use the character's UTF-8 representation to avoid this confusion. For example, if a URL contains a space, you might want to encode the space as \x20 to avoid confusing certain browsers and web server software.
Below are examples of URLs, form field names, and safe object expressions that contain non-ASCII characters that must be entered as PCRE-format regular expressions to be included in the application firewall configuration. Each example shows the actual URL, field name, or expression string first, followed by a PCRE-format regular expression for it.

- A URL containing extended ASCII characters.

  Actual URL: http://www.josénuñez.com

  Encoded URL: ^http://www[.]jos\xC3\xA9nu\xC3\xB1ez[.]com$

- Another URL containing extended ASCII characters.

  Actual URL: http://www.example.de/trömso.html

  Encoded URL: ^http://www[.]example[.]de/tr\xC3\xB6mso[.]html$

- A form field name containing extended ASCII characters.

  Actual Name: nome_do_usuário

  Encoded Name: ^nome_do_usu\xC3\xA1rio$

- A safe object expression containing extended ASCII characters.

  Unencoded Expression [A-Z]{3,6}¥[1-9][0-9]{6,6}

  Encoded Expression: [A-Z]{3,6}\xC2\xA5[1-9][0-9]{6,6}

You can find a number of tables that include the entire Unicode character set and matching UTF-8 encodings on the Internet. A useful web site that contains this information is located at the following URL:

http://www.utf8-chartable.de/unicode-utf8-table.pl
For the characters in the table on this web site to display correctly, you must have an appropriate Unicode font installed on your computer. If you do not, the visual display of the character may be in error. Even if you do not have an appropriate font installed to display a character, however, the description and the UTF-8 and UTF-16 codes on this set of web pages will be correct.

Inverted PCRE Expressions

In addition to matching content that contains a pattern, you can match content that does not contain a pattern by using an inverted PCRE expression. To invert an expression, you simply include an exclamation point (!) followed by white space as the first character in the expression.

Note: If an expression consists only of an exclamation point with nothing following, the exclamation point is treated as a literal character, not syntax indicating an inverted expression.
The following application firewall commands support inverted PCRE expressions:

- Start URL (URL)
- Deny URL (URL)
- Form Field Consistency (form action URL)
- Cookie Consistency (form action URL)
- Cross Site Request Forgery (CSRF) (form action URL)
- HTML Cross-site Scripting (form action URL)
- Field Format (form action URL)
- Field Type (type)
- Confidential Field (URL)

Note: If the security check contains an isRegex flag or check box, it must be set to YES or checked to enable regular expressions in the field. Otherwise the contents of that field are treated as literal and no regular expressions (inverted or not) are parsed.

Disallowed Names for Application Firewall Profiles

The following names are assigned to built-in actions and profiles on the NetScaler appliance, and cannot be used as names for a user-created application firewall profile.

- AGRESSIVE
- ALLOW
- BASIC
- CLIENTAUTH
- COMPRESS
- CSSMINIFY
- DEFLATE
- DENY
- DNS-NOP
- DROP
- GZIP
- HTMLMINIFY
- IMGOPTIMIZE
- JSMINIFY
- MODERATE
- NOCLIENTAUTH
- NOCOMPRESS
- NONE
- NOOP
- NOREWRITE
- RESET
- SETASLEARNNSLOG_ACT

- SETNSLOGPARAMS_ACT
- SETSYSLOGPARAMS_ACT
- SETTMSESSPARAMS_ACT
- SETVPNPARAMS_ACT
- SET_PREAUTHPARAMS_ACT
- default_DNS64_action
- dns_default_act_Cachebypass
- dns_default_act_Drop
- nshttp_default_profile
- nshttp_default_strict_validation
- nstcp_default_Mobile_profile
- nstcp_default_XA_XD_profile
- nstcp_default_profile
- nstcp_default_tcp_interactive_stream
- nstcp_default_tcp_lan
- nstcp_default_tcp_lan_thin_stream
- nstcp_default_tcp_lfp
- nstcp_default_tcp_lfp_thin_stream
- nstcp_default_tcp_lnp
- nstcp_default_tcp_lnp_thin_stream
- nstcp_internal_apps

Policy Labels

Nov 19, 2013

A policy label consists of a set of policies, other policy labels, and virtual server-specific policy banks. The application firewall evaluates each policy bound to the policy label in order of priority. If the policy matches, it filters the connection as specified in the associated profile. Then it does whatever the Goto parameter specifies, which can be to terminate policy evaluation, go to the next policy, or go to the policy with the specified priority. If the Invoke parameter is set, it terminates processing of the current policy label and begins to process the specified policy label or virtual server.

To create an application firewall policy label by using the command line

At the command prompt, type the following commands:

- add appfw policylabel <labelName> http_req
- save ns config

Example

The following example creates a policy label named **policylbl1**.

```
add appfw policylabel policylbl1 http_req
save ns config
```

To bind a policy to a policy label by using the command line

At the command prompt, type the following commands:

- bind appfw policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]
- save ns config

Example

The following example binds the policy **policy1** to the policy label **policylbl1** with a priority of 1.

```
bind appfw policylabel policylbl1 policy1 1
save ns config
```

To configure an application firewall policy label by using the configuration utility

1. Navigate to Security > Application Firewall > Policy Labels.
2. In the details pane, do one of the following:
   - To add a new policy label, click Add.
   - To configure an existing policy label, select the policy label and the click Open.

   The Create Application Firewall Policy Label or the Configure Application Firewall Policy Label dialog box opens. The dialog boxes are nearly identical.
3. If you are creating a new policy label, in the Create Application Firewall Policy Label dialog box, type a name for your new policy label.

   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) symbols.

4. Select Insert Policy to insert a new row and display a drop-down list with all existing application firewall policies.
5. Select the policy you want to bind to the policy label, or select New Policy to create a new policy and follow the instructions in To create and configure a policy by using the configuration utility. The policy that you selected or created is inserted into the list of globally bound application firewall policies.
6. Make any additional adjustments.
   - To modify the policy priority, click the field to enable it, and then type a new priority. You can also select Regenerate Priorities to renumber the priorities evenly.
   - To modify the policy expression, double click that field to open the Configure Application Firewall Policy dialog box, where you can edit the policy expression.
   - To set the Goto Expression, double click field in the Goto Expression column heading to display the drop-down list, where you can choose an expression.
   - To set the Invoke option, double click field in the Invoke column heading to display the drop-down list, where you can choose an expression
7. Repeat steps 5 through 7 to bind any additional application firewall policies you want to the policy label.
8. Click Create or OK, and then click Close. A message appears in the status bar, stating that you have successfully created or modified the policy label.

Policies

Mar 28, 2012

The application firewall uses two types of policies: firewall policies and auditing policies. Firewall policies control which traffic is sent to the application firewall. Auditing policies control the log server to which application firewall logs are sent.

Firewall policies can be complex because the policy rule can consist of multiple expressions in the NetScaler expressions language, which is a full-fledged object oriented programming language capable of defining with extreme precision exactly which connections to filter. Because firewall policies operate within the context of the application firewall, they must meet certain criteria that are connected to how the application firewall functions and what traffic is appropriately filtered by it. As long as you keep these criteria in mind, however, firewall policies are similar to policies for other NetScaler features. The instructions here do not attempt to cover all aspects of writing firewall policies, but only provide an introduction to policies and cover those criteria that are unique to the application firewall.

Auditing policies are simple because the policy rule is always ns_true. You need only specify the log server that you want to send logs to, the logging levels that you want to use, and a few other criteria that are explained in detail.

Firewall Policies

Oct 01, 2013

A firewall policy is a rule associated with a profile. The rule is an expression or group of expressions that defines the types of request/response pairs that the application firewall is to filter by applying the profile. Firewall policy expressions are written in the NetScaler expressions language, an object-oriented programming language with special features to support specific NetScaler functions. The profile is the set of actions that the application firewall is to use to filter request/response pairs that match the rule.

Firewall policies enable you to assign different filtering rules to different types of web content. Not all web content is alike. A simple web site that uses no complex scripting and accesses and handles no private data might require only the level of protection provided by a profile created with basic defaults. Web content that contains JavaScript-enhanced web forms or accesses an SQL database probably requires more tailored protection. You can create a different profile to filter that content, and create a separate firewall policy that can determine which requests are attempting to access that content. You then associate the policy expression with a profile you created and globally bind the policy to put it into effect.

The application firewall processes only HTTP connections, and therefore uses a subset of the overall NetScaler expressions language. The information here is limited to topics and examples that are likely to be useful when configuring the application firewall. Following are links to additional information and procedures for firewall policies:

- For procedures that explain how to create and configure a policy, see "Creating and Configuring Application Firewall Policies."
- For a procedure that explains in detail how to create a policy rule (expression), see "To create or configure an Application Firewall rule (expression)."
- For a procedure that explains how to use the Add Expression dialog box to create a policy rule, see "To add a firewall rule (expression) by using the Add Expression dialog box."
- For a procedure that explains how to view the current bindings for a policy, see "Viewing a Firewall Policy's Bindings."
- For procedures that explain how to bind an application firewall policy, see "Binding Application Firewall Policies."
- For detailed information about the NetScaler expressions language, see "Policies and Expressions."

Creating and Configuring Application Firewall Policies

Jun 12, 2014

A firewall policy consists of two elements: a *rule*, and an associated *profile*. The rule selects the HTTP traffic that matches the criteria that you set, and sends that traffic to the application firewall for filtering. The profile contains the filtering criteria that the application firewall uses.

The policy rule consists of one or more expressions in the NetScaler expressions language. The NetScaler expressions syntax is a powerful, object-oriented programming language that enables you to precisely designate the traffic that you want to process with a specific profile. For users who are not completely familiar with the NetScaler expressions language syntax, or who prefer to configure their NetScaler appliance by using a web-based interface, the configuration utility provides two tools: the Prefix menu and the Add Expression dialog box. Both help you to write expressions that select exactly the traffic that you want to process. Experienced users who are thoroughly familiar with the syntax may prefer to use the NetScaler command line to configure their NetScaler appliances.

Note: In addition to the default expressions syntax, for backward compatibility the NetScaler operating system supports the NetScaler classic expressions syntax on NetScaler Classic and nCore appliances and virtual appliances. Classic expressions are not supported on NetScaler Cluster appliances and virtual appliances. Current NetScaler users who want to migrate existing configurations to the NetScaler Cluster must migrate any policies that contain classic expressions to the default expressions syntax.
For detailed information about the NetScaler expressions languages, see "Policies and Expressions."

You can create a firewall policy by using the configuration utility or the NetScaler command line.

To create and configure a policy by using the command line interface

At the command prompt, type the following commands:

- add appfw policy <name> <rule> <profileName>
- save ns config

Example

The following example adds a policy named **pl-blog**, with a rule that intercepts all traffic to or from the host **blog.example.com**, and associates that policy with the profile **pr-blog**. This is an appropriate policy to protect a blog hosted on a specific hostname.

```
add appfw policy pl-blog "HTTP.REQ.HOSTNAME.DOMAIN.EQ("blog.example.com")" pr-blog
```

To create and configure a policy by using the configuration utility

1. Navigate to Security > Application Firewall > Policies.
2. In the details pane, do one of the following:
   - To create a new firewall policy, click Add. The Create Application Firewall Policy is displayed.
   - To edit an existing firewall policy, select the policy, and then click Open.

   The Create Application Firewall Policy or Configure Application Firewall Policy is displayed.
3. If you are creating a new firewall policy, in the Create Application Firewall Policy dialog box, Policy Name text box, type a name for your new policy.
   The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 128 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) symbols.

If you are configuring an existing firewall policy, this field is read-only. You cannot modify it.

4. Select the profile that you want to associate with this policy from the Profile drop-down list. You can create a new profile to associate with your policy by clicking New, and you can modify an existing profile by clicking Modify.
5. In the Expression text area, create a rule for your policy.
   - You can type a rule directly into the text area.
   - You can click Prefix to select the first term for your rule, and follow the prompts. See "To Create an Application Firewall Rule (Expression)" for a complete description of this process.
   - You can click Add to open the Add Expression dialog box, and use it to construct the rule. See "The Add Expression Dialog Box" for a complete description of this process.
6. Click Create or OK, and then click Close.

To create or configure an Application Firewall rule (expression)

The policy rule, also called the *expression*, defines the web traffic that the application firewall filters by using the profile associated with the policy. Like other NetScaler policy rules (or *expressions*), application firewall rules use NetScaler expressions syntax. This syntax is powerful, flexible, and extensible. It is too complex to describe completely in this set of instructions. You can use the following procedure to create a simple firewall policy rule, or you can read it as an overview of the policy creation process.

1. If you have not already done so, navigate to the appropriate location in the Application Firewall wizard or the NetScaler configuration utility to create your policy rule:
   - If you are configuring a policy in the Application Firewall wizard, in the navigation pane, click Application Firewall, then in the details pane click Application Firewall Wizard, and then navigate to the Specify Rule screen.
   - If you are configuring a policy manually, in the navigation pane, expand Application Firewall, then Policies, and then Firewall. In the details pane, to create a new policy, click Add. To modify an existing policy, select the policy, and then click Open.
2. On the Specify Rule screen, the Create Application Firewall Profile dialog box, or the Configure Application Firewall Profile dialog box, click Prefix, and then choose the prefix for your expression from the drop-down list. Your choices are:
   - **HTTP.** The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol.
   - **SYS.** The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   - **CLIENT.** The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.
   - **SERVER.** The computer to which the request was sent. Choose this if you want to examine some aspect of the recipient of the request.

   After you choose a prefix, the application firewall displays a two-part prompt window that displays the possible next choices at the top, and a brief explanation of what the selected choice means at the bottom.

3. Choose your next term.
   If you chose HTTP as your prefix, your only choice is REQ, which specifies the Request/Response pair. (The application firewall operates on the request and response as a unit instead of on each separately.) If you chose another prefix, your choices are more varied. For help on a specific choice, click that choice once to display information about it in the lower prompt window.

   When you have decided which term you want, double-click it to insert it into the Expression window.

4. Type a period after the term you just chose. You are then prompted to choose your next term, as described in the

previous step. When a term requires that you type a value, fill in the appropriate value. For example, if you choose HTTP.REQ.HEADER(""), type the header name between the quotation marks.

5. Continue choosing terms from the prompts and filling in any values that are needed, until your expression is finished. Following are some examples of expressions for specific purposes.

   - **Specific web host.** To match traffic from a particular web host:

     HTTP.REQ.HEADER("Host").EQ("shopping.example.com")
     For shopping.example.com, substitute the name of the web host that you want to match.

   - **Specific web folder or directory.** To match traffic from a particular folder or directory on a Web host:

     HTTP.REQ.URL.STARTSWITH("https//www.example.com/folder")
     For www.example.com, substitute the name of the web host. For folder, substitute the folder or path to the content that you want to match. For example, if your shopping cart is in a folder called /solutions/orders, you substitute that string for folder.

   - **Specific type of content: GIF images.** To match GIF format images:

     HTTP.REQ.URL.ENDSWITH(".gif")
     To match other format images, substitute another string in place of .gif.

   - **Specific type of content: scripts.** To match all CGI scripts located in the CGI-BIN directory:

     HTTP.REQ.URL.STARTSWITH("https//www.example.com/CGI-BIN")
     To match all JavaScripts with .js extensions:

     HTTP.REQ.URL.ENDSWITH(".js")
     For more information about creating policy expressions, see "Policies and Expressions."

   Note: If you use the command line to configure a policy, remember to escape any double quotation marks within NetScaler expressions. For example, the following expression is correct if entered in the configuration utility:
   HTTP.REQ.HEADER("Host").EQ("shopping.example.com")
   If entered at the command line, however, you must type this instead:

   HTTP.REQ.HEADER(\"Host\").EQ(\"shopping.example.com\")

To add a firewall rule (expression) by using the Add Expression dialog box

The Add Expression dialog box (also referred to as the Expression Editor) helps users who are not familiar with the NetScaler expressions language to construct a policy that matches the traffic that they want to filter.

1. If you have not already done so, navigate to the appropriate location in the Application Firewall wizard or the NetScaler configuration utility:
   - If you are configuring a policy in the Application Firewall wizard, in the navigation pane, click Application Firewall, then in the details pane click Application Firewall Wizard, and then navigate to the Specify Rule screen.
   - If you are configuring a policy manually, in the navigation pane, expand Application Firewall, then Policies, and then Firewall. In the details pane, to create a new policy, click Add. To modify an existing policy, select the policy, and then click Open.
2. On the Specify Rule screen, in the Create Application Firewall Profile dialog box, or in the Configure Application Firewall Profile dialog box, click Add.

3. In the Add Expression dialog box, in the Construct Expression area, in the first list box, choose one of the following prefixes:
   - **HTTP.** The HTTP protocol. Choose this if you want to examine some aspect of the request that pertains to the HTTP protocol. The default choice.
   - **SYS.** The protected Web site(s). Choose this if you want to examine some aspect of the request that pertains to the recipient of the request.
   - **CLIENT.** The computer that sent the request. Choose this if you want to examine some aspect of the sender of the request.
   - **SERVER.** The computer to which the request was sent. Choose this if you want to examine some aspect of the recipient of the request.
4. In the second list box, choose your next term. The available terms differ depending on the choice you made in the previous step, because the dialog box automatically adjusts the list to contain only those terms that are valid for the context. For example, if you selected HTTP in the previous list box, the only choice is REQ, for requests. Because the application firewall treats requests and associated responses as a single unit and filters both, you do not need to specific responses separately. After you choose your second term, a third list box appears to the right of the second. The Help window displays a description of the second term, and the Preview Expression window displays your expression.
5. In the third list box, choose the next term. A new list box appears to the right, and the Help window changes to display a description of the new term. The Preview Expression window updates to display the expression as you have specified it to that point.
6. Continue choosing terms, and when prompted filling in arguments, until your expression is complete. If you make a mistake or want to change your expression after you have already selected a term, you can simply choose another term. The expression is modified, and any arguments or additional terms that you added after the term that you modified are cleared.
7. When you have finished constructing your expression, click OK to close the Add Expression dialog box. Your expression is inserted into the Expression text area.

Binding Application Firewall Policies

Jun 12, 2014

After you have configured your application firewall policies, you bind them to Global or a bind point to put them into effect. After binding, any request or response that matches an application firewall policy is transformed by the profile associated with that policy.

When you bind a policy, you assign a priority to it. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer. In the NetScaler OS, policy priorities work in reverse order - the higher the number, the lower the priority.

Because the application firewall feature implements only the first policy that a request matches, not any additional policies that it might also match, policy priority is important for achieving the results that you intend. If you give your first policy a low priority (such as 1000), you configure the application firewall to perform it only if other policies with a higher priority do not match a request. If you give your first policy a high priority (such as 1), you configure the application firewall to perform it first, and skip any other policies that might also match. You can leave yourself plenty of room to add other policies in any order, without having to reassign priorities, by setting priorities with intervals of 50 or 100 between each policy when you bind your policies.

For more information about binding policies on the NetScaler appliance, see "Policies and Expressions."

To bind an application firewall policy by using the command line interface

At the command prompt, type the following commands:

- bind appfw global <policyName> <priority>
- save ns config

Example

The following example binds the policy named **pl-blog** and assigns it a priority of **10**.

```
bind appfw global pl-blog 10
save ns config
```

To bind an application firewall policy by using the configuration utility

1. Do one of the following:
   - Navigate to Security > Application Firewall, and in the details pane, click Application Firewall policy manager.
   - Navigate to Security > Application Firewall > Policies > Firewall Policies, and in the details pane, click Policy Manager.
2. In the Application Firewall Policy Manager dialog, choose the bind point to which you want to bind the policy from the drop-down list. The choices are:
   - **Override Global.** Policies that are bound to this bind point process all traffic from all interfaces on the NetScaler appliance, and are applied before any other policies.
   - **LB Virtual Server.** Policies that are bound to a load balancing virtual server are applied only to traffic that is processed by that load balancing virtual server, and are applied before any Default Global policies. After selecting LB Virtual Server, you must also select the specific load balancing virtual server to which you want to bind this policy.
   - **CS Virtual Server.** Policies that are bound to a content switching virtual server are applied only to traffic that is processed by that content switching virtual server, and are applied before any Default Global policies. After selecting CS Virtual Server, you must also select the specific content switching virtual server to which you want to bind this

policy.

- **Default Global.** Policies that are bound to this bind point process all traffic from all interfaces on the NetScaler appliance.
- **Policy Label.** Policies that are bound to a policy label process traffic that the policy label routes to them. The policy label controls the order in which policies are applied to this traffic.
- **None.** Do not bind the policy to any bind point.

3. Select Insert Policy to insert a new row and display a drop-down list with all available, unbound application firewall policies.
4. Select the policy you want to bind, or select New Policy to create a new policy. The policy that you selected or created is inserted into the list of globally bound application firewall policies.
5. Make any additional adjustments to the binding.
   - To modify the policy priority, click the field to enable it, and then type a new priority. You can also select Regenerate Priorities to renumber the priorities evenly.
   - To modify the policy expression, double click that field to open the Configure Application Firewall Policy dialog box, where you can edit the policy expression.
   - To set the Goto Expression, double click field in the Goto Expression column heading to display the drop-down list, where you can choose an expression.
   - To set the Invoke option, double click field in the Invoke column heading to display the drop-down list, where you can choose an expression
6. Repeat steps 3 through 6 to add any additional application firewall policies you want to globally bind.
7. Click OK. A message appears in the status bar, stating that the policy has been successfully bound.

Viewing a Firewall Policy's Bindings

Jun 12, 2014

You can quickly check to determine what bindings are in place for any firewall policy by viewing the bindings in the configuration utility.

To view bindings for an application firewall policy

1. Navigate to Security > Application Firewall > Policies > Firewall Policies
2. In the details pane, select the policy that you want to check, and then click Show Bindings. The Binding Details for Policy: Policy message box is displayed, with a list of bindings for the selected policy.
3. Click Close.

Supplemental Information about Application Firewall Policies

Jun 07, 2012

Following is supplemental information about particular aspects of application firewall policies that system administrators who manage the application firewall might need to know.

Correct but Unexpected Behavior

Web application security and modern web sites are complex. In a number of scenarios, a NetScaler policy might cause the application firewall to behave differently in certain situations than a user who is familiar with policies would normally expect. Following are a number of cases where the application firewall may behave in an unexpected fashion.

- **Request with a missing HTTP Host header and an absolute URL.** When a user sends a request, in the majority of cases the request URL is relative. That is, it takes as its starting point the Referer URL, the URL where the user's browser is located when it sends the request. If a request is sent without a Host header, and with a relative URL, the request is normally blocked both because it violates the HTTP specification and because a request that fails to specify the host could under some circumstances constitute an attack. If a request is sent with an absolute URL, however, even if the Host header is missing, the request bypasses the application firewall and is forwarded to the web server. Although such a request violates the HTTP specification, it poses no possible threat because an absolute URL contains the host.

Auditing Policies

Jun 12, 2014

Auditing policies determine the messages that are generated and logged during an Application Firewall session. These messages are logged in SYSLOG format to the local NSLOG server or to an external logging server. Different types of messages are logged on the basis of the level of logging selected.

To create an auditing policy, you must first create either an NSLOG server or a SYSLOG server. After specifying the server, you create the policy and specify the type of log and the server to which logs are sent.

To create an auditing server by using the command line interface

You can create two different types of auditing server: an NSLOG server or a SYSLOG server. The command names are different, but the parameters for the commands are the same.

To create an auditing server, at the NetScaler command prompt, type the following commands:

- add audit syslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> ... [-dateFormat (**MMDDYYYY** | **DDMMYYYY**)] [-logFacility <logFacility>] [-tcp (**NONE** | **ALL**)] [-acl (**ENABLED** | **DISABLED**)] [-timeZone (**GMT_TIME** | **LOCAL_TIME**)] [-userDefinedAuditlog (**YES** | **NO**)] [-appflowExport (**ENABLED** | **DISABLED**)]
- save ns config

Example

The following example creates a syslog server named syslog1 at IP 10.124.67.91, with loglevels of emergency, critical, and warning, log facility set to LOCAL1, that logs all TCP connections:

```
add audit syslogAction syslog1 10.124.67.91 -logLevel emergency critical warning -logFacility
LOCAL1 -tcp ALL
save ns config
```

To modify or remove an auditing server by using the command line interface

- To modify an auditing server, type the set audit <type> command, the name of the auditing server, and the parameters to be changed, with their new values.
- To remove an auditing server, type the rm audit <type> command and the name of the auditing server.

Example

The following example modifies the syslog server named syslog1 to add errors and alerts to the log level:

```
set audit syslogAction syslog1 10.124.67.91 -logLevel emergency critical warning alert error
-logFacility LOCAL1 -tcp ALL
save ns config
```

To create or configure an auditing server by using the configuration utility

1. Navigate to Security > Application Firewall > Policies > Auditing.
2. In the details pane, click the Server tab.
3. Do one of the following:
   - To add a new auditing server, click Add.
   - To modify an existing auditing server, select the server, and then click Open.

4. In the Create Auditing Server or Configure Auditing Server dialog box, set the following parameters:
   - Name
   - Auditing Type
   - IP Address
   - Port
   - Log Levels
   - Log Facility
   - TCP Logging
   - ACL Logging
   - User-Configurable Log Messages
   - AppFlow Logging
   - Date Format
   - Time Zone
5. Click Create or OK.

To create an auditing policy by using the command line interface

You can create an NSLOG policy or a SYSLOG policy. The type of policy must match the type of server. The command names for the two types of policy are different, but the parameters for the commands are the same.

At the command prompt, type the following commands:

- add audit syslogPolicy <name> [-rule <expression>] [-action <string>]
- save ns config

Example

The following example creates a policy named syslogP1 that logs application firewall traffic to a syslog server named syslog1.

```
add audit syslogPolicy syslogP1 -rule "ns_true" -action syslog1
save ns config
```

To configure an auditing policy by using the command line interface

At the command prompt, type the following commands:

- set audit syslogPolicy <name> [-rule <expression>] [-action <string>]
- save ns config

Example

The following example modifies the policy named syslogP1 to log application firewall traffic to a syslog server named syslog2.

```
set audit syslogPolicy syslogP1 -rule "ns_true" -action syslog2
save ns config
```

To configure an auditing policy by using the configuration utility

1. Navigate to Security > Application Firewall > Policies > Auditing.
2. In the details pane, do one of the following:
   - To add a new policy, click Add.

- To modify an existing policy, select the policy, and then click Open.

3. In the Create Auditing Policy or Configure Auditing Policy dialog box, set the following parameters:
   - Name
   - Auditing Type
   - Server
4. Click Create or OK.

Imports

Mar 28, 2012

Several application firewall features make use of external files that you upload to the application firewall when configuring it. Using the configuration utility, you manage those files in the Imports pane, which has four tabs corresponding to the four types of files you can import: HTML error objects, XML error objects, XML schemas, and Web Services Description Language (WSDL) files. Using the NetScaler command line, you can import these types of files, but you cannot export them.

HTML Error Object

When a user's connection to an HTML or Web 2.0 page is blocked, or a user asks for a non-existent HTML or Web 2.0 page, the application firewall sends an HTML-based error response to the user's browser. When configuring which error response the application firewall should use, you have two choices:

- You can configure a redirect URL, which can be hosted on any Web server to which users also have access. For example, if you have a custom error page on your Web server, `404.html`, you can configure the application firewall to redirect users to that page when a connection is blocked.
- You can configure an HTML error object, which is an HTML-based Web page that is hosted on the application firewall itself. If you choose this option, you must upload the HTML error object to the application firewall. You do that in the Imports pane, on the HTML Error Object tab.

The error object must be a standard HTML file that contains no non-HTML syntax except for application firewall error object customization variables. It cannot contain any CGI scripts, server-parsed code, or PHP code. The customization variables enable you to embed troubleshooting information in the error object that the user receives when a request is blocked. While most requests that the application firewall blocks are illegitimate, even a properly configured application firewall can occasionally block legitimate requests, especially when you first deploy it or after you make significant changes to your protected Web sites. By embedding information in the error page, you provide the user with the information that he or she needs to give to the technical support person so that any issues can be fixed.

The application firewall error page customization variables are:

- `${NS_TRANSACTION_ID}`. The transaction ID that the application firewall assigned to this transaction.
- `${NS_APPFW_SESSION_ID}`. The application firewall session ID.
- `${NS_APPFW_VIOLATION_CATEGORY}`. The specific application firewall security check or rule that was violated.
- `${NS_APPFW_VIOLATION_LOG}`. The detailed error message associated with the violation.
- `${COOKIE("<CookieName>")}`. The contents of the specified cookie. For <CookieName>, substitute the name of the specific cookie that you want to display on the error page. If you have multiple cookies whose contents you want to display for troubleshooting, you can use multiple instances of this customization variable, each with the appropriate cookie name.
  Note: If you have blocking enabled for the Cookie Consistency Check, any blocked cookies are not displayed on the error page because the application firewall blocks them.

To use these variables, you embed them in the HTML or XML of the error page object as if they were an ordinary text string. When the error object is displayed to the user, for each customization variable the application firewall substitutes the information to which the variable refers. An example HTML error page that uses custom variables is shown below.

```
<!doctype html public "-//w3c//dtd html 4.0//en"> <html> <head> <title>Page Not Accessible</title> </head> <body> <h1>Page Not Accessible</h1> <p>The pag
```

To use this error page, copy it into a text or HTML editor. Substitute the appropriate local information for the following variables, which are enclosed in square brackets to distinguish them from the NetScaler variables. (Leave those unchanged.):

- **[homePage]**. The URL for your web site's home page.
- **[helpDeskEmailAddress]**. The email address that you want users to use to report blocking incidents.
- **[helpDeskPhoneNumber]**. The phone number that you want users to call to report blocking incidents.
- **[cookieName]**. The name of the cookie whose contents you want to display on the error page.

XML Error Object

When a user's connection to an XML page is blocked, or a user asks for a nonexistent XML application, the application firewall sends an XML-based error response to the user's browser. You configure the error response by uploading an XML-based error page to the application firewall in the Imports Pane, on the XML Error Object tab. All XML error responses are hosted on the application firewall. You cannot configure a redirect URL for XML applications.

Note: You can use the same customization variables in an XML error object as in an HTML error object.

XML Schema

When the application firewall performs a validation check on a user's request for an XML or Web 2.0 application, it can validate the request against the XML schema or design type document (DTD) for that application and reject any request that does not follow the schema or DTD. Both an XML schema and a DTD are standard XML configuration files that describe the structure of a specific type of XML document.

WSDL

When the application firewall performs a validation check on a user's request for an XML SOAP-based web service, it can validate the request against the web services type definition (WSDL) file for that web service. A WSDL file is a standard XML SOAP configuration file that defines the elements of a specific XML SOAP web service.

Importing and Exporting Files

Oct 02, 2014

You can import HTML or XML error objects, XML schemas, DTDs, and WSDLs to the application firewall by using the configuration utility or the command line. You can edit any of these files in a web-based text area after importing them, to make small changes directly on the NetScaler ADC instead of having to make them on your computer and then reimport them. Finally, you can export any of these files to your computer, or delete any of these files, by using the configuration utility.

Note: You cannot delete or export an imported file by using the command line.

To import a file by using the command line interface

At the command prompt, type the following commands:

- import appfw htmlerrorpage <src> <name>
- save ns config

Example

The following example imports an HTML error object from a file named error.html and assigns it the name HTMLError.

```
import htmlerrorpage error.html HTMLError
save ns config
```

To import a file by using the configuration utility

Before you attempt to import an XML schema, DTD, or WSDL file, or an HTML or XML error object from a network location, verify that the NetScaler ADC can connect to the Internet or LAN computer where the file is located. Otherwise, you cannot import the file or object.

1. Navigate to Security > Application Firewall > Imports.
2. Navigate to Application Firewall > Imports.
3. In the Application Firewall Imports pane, select the tab for the type of file you want to import, and then click Add.
   The tabs are HTML Error Page, XML Error Page, XML Schema or WSDL. The upload process is identical on all four tabs from the user point of view.
4. Fill in the dialog fields.
   - **Name**—A name for the imported object.
   - **Import From**—Choose the location of the HTML file, XML file, XML schema or WSDL that you want to import in the drop-down list:
     - **URL:** A web URL on a website accessible to the ADC.
     - **File:** A file on a local or networked hard disk or other storage device.
     - **Text:** Type or paste the text of the custom response directly into a text field in the configuration utility.

     The third text box changes to the appropriate value. The three possible values are provided below.
   - **URL**—Type the URL into the text box.
   - **File**—Type the path and filename to the HTML file directly, or click Browse and browse to the HTML file.
   - **Text**—The third field is removed, leaving a blank space.
5. Click Continue. The File Contents dialog is displayed. If you chose URL or File, the File Contents text box contains the HTML file that you specified. If you chose Text, the File Contents text box is empty.

6. If you chose Text, type or copy and paste the custom response HTML that you want to import.
7. Click Done.
8. To delete an object, select the object, and then click Delete.

To export a file by using the configuration utility

Before you attempt to export an XML schema, DTD, or WSDL file, or an HTML or XML error object, verify that the application firewall appliance can access the computer where the file is to be saved. Otherwise, you cannot export the file.

1. Navigate to Security > Application Firewall > Imports.
2. In the Application Firewall Imports pane, select the tab for the type of file you want to export.
   The export process is identical on all four tabs from the user point of view.
3. Select the file that you want to export.
4. Click Export.
5. In the dialog box, choose Save File and click OK.
6. In the Browse dialog box, navigate to the local file system and directory where you want to save the exported file, and click Save.

To edit an HTML or XML Error Object in the configuration utility

You edit the text of HTML and XML error objects in the configuration utility without exporting and then reimporting them.

1. Navigate to Security > Application Firewall > Imports, and then select the tab for the type of file that you want to modify.
2. Navigate to Application Firewall > Imports, and then select the tab for the type of file that you want to modify.
3. Select the file that you want to modify, and then click Open.
   The text of the HTML or XML error object is displayed in a browser text area. You can modify the text by using the standard browser-based editing tools and methods for your browser.

   Note: The edit window is designed to allow you to make minor changes to your HTML or XML error object. To make extensive changes, you may prefer to export the error object to your local computer and use standard HTML or XML web page editing tools.
4. Click OK, and then click Close.

Global Configuration

Sep 12, 2013

The application firewall global configuration affects all profiles and policies. The Global Configuration items are:

- **Engine Settings.** A collection of global settings—session cookie name, session time-out, maximum session lifetime, logging header name, undefined profile, default profile, and import size limit—that pertain to all connections that the application firewall processes, rather than to a specific subset of connections.
- **Confidential Fields.** A set of form fields in web forms that contain sensitive information that should not be logged to the application firewall logs. Form fields such as password fields on a logon page or credit card information on a shopping cart checkout form are normally designated as confidential fields.
- **Field Types.** The list of web form field types used by the Field Formats security check. Each of these field types is defined by a PCRE-compliant regular expression that defines the type of data and the minimum/maximum length of data that should be allowed in that type of form field.
- **XML Content Types.** The list of content types recognized as XML and subjected to XML-specific security checks. Each of these content types is defined by a PCRE-compliant regular expression that defines the exact MIME type assigned to that content.
- **JSON Content Types.** The list of content types recognized as JSON and subjected to JSON-specific security checks. Each of these content types is defined by a PCRE-compliant regular expression that defines the exact MIME type assigned to that content.

Engine Settings

Oct 07, 2014

The engine settings affect all requests and responses that the application firewall processes. They include the following items:

- **Cookie name**—The name of the cookie that stores the NetScaler session ID.
- **Session timeout**—The maximum inactive period allowed. If a user session shows no activity for this length of time, the session is terminated and the user is required to reestablish it by visiting a designated start page.
- **Cookie post-encrypt prefix**—The string that precedes the encrypted portion of any encrypted cookies.
- **Maximum session lifetime**—The maximum amount of time, in seconds, that a session is allowed to remain live. After this period is reached, the session is terminated and the user is required to reestablish it by visiting a designated start page. This setting cannot be less then the session timeout. To disable this setting, so that there is no maximum session lifetime, set the value to zero (0).
- **Logging header name**—The name of the HTTP header that holds the Client IP, for logging.
- **Undefined profile**—The profile applied when the corresponding policy action evaluates as undefined.
- **Default profile**—The profile applied to connections that do not match a policy.
- **Import size limit**—The maximum cumulative total byte count of all files imported to the ADC, including signatures, WSDLs, schemas, HTML and XML error pages. During an import, if the size of the imported object would cause the cumulative total sizes of all imported files to exceed the configured limit, the import operation fails and the ADC displays the following error message: *ERROR: Import failed - exceeding the configured total size limit on the imported objects*.
- **Learn message rate limit**—The maximum number of requests and responses per second that the learning engine is to process. Any additional requests or responses over this limit are not sent to the learning engine.
-
- **Log malformed request**—Enable logging of malformed HTTP requests.
- **Use configurable secret key**—Use a configurable secret key for application firewall operations.
- **Manage learned data**—Remove all learned data from the application firewall. Restarts the learning process by collecting fresh data.
- **Signature auto-update settings**—Enable/disable automatic updating of the application firewall default signatures.
- **Signature auto-update URL**—Remove all learned data from the application firewall. Restarts the learning process by collecting fresh data.

Normally, the default values for the application firewall settings are correct. If the default settings cause a conflict with other servers or cause premature disconnection of your users, however, you might need to modify them.

To configure engine settings by using the command line interface

At the command prompt, type the following commands:

- set appfw settings [-sessionCookieName <name>] [-sessionTimeout <positiveInteger>] [-sessionLifetime <positiveInteger>] [-clientIPLoggingHeader <headerName>] [-undefaction <profileName>] [-defaultProfile <profileName>] [-importSizeLimit <positiveInteger>] [-logMalformedReq (ON | OFF)] [-signatureAutoUpdate (ON | OFF)] [-signatureUrl <expression>] [-cookiePostEncryptPrefix <string>] [-useConfigurableSecretKey (ON | OFF)] [-learnRateLimit <positiveInteger>]
- save ns config

Example

```
set appfw settings -sessionCookieName citrix-appfw-id -sessionTimeout 3600
-sessionLifetime 14400 -clientIPLoggingHeader NS-AppFW-Client-IP -undefaction APPFW_RESET
-defaultProfile APPFW_RESET -importSizeLimit 4096
save ns config
```

To configure engine settings by using the configuration utility

1. Navigate to Security > Application Firewall
2. In the details pane, click Change Engine Settings.
3. In the Application Firewall Engine Settings dialog box, set the following parameters:
   - Cookie Name
   - Session Timeout
   - Cookie Post Encrypt Prefix
   - Maximum Session Lifetime
   - Logging Header Name
   - Undefined Profile
   - Default Profile
   - Import Size Limit
   - Learn Messages Rate Limit
   -
   - Entity Decoding
   - Log Malformed Request
   - Use Configurable Secret Key
   - Manage Learned Data
   - Signatures Auto Update
   - Signatures Update URL
4. Click OK.

Confidential Fields

Jun 12, 2014

You can designate web-form fields as confidential to protect the information users type into them. Normally, any information a user types into a web form on one of your protected web servers is logged in the NetScaler logs. The information typed into a web-form field designated as confidential, however, is not logged. That information is saved only where the web site is configured to save such data, normally in a secure database.

Common types of information that you may want to protect with a confidential field designation include:

- Passwords
- Credit card numbers, validation codes, and expiration dates
- Social security numbers
- Tax ID numbers
- Home addresses
- Private telephone numbers

In addition to being good practice, proper use of confidential field designations may be necessary for PCI-DSS compliance on ecommerce servers, HIPAA compliance on servers that manage medical information in the United States, and compliance with other data protection standards.

Important: In the following two cases, the Confidential Field designation does not function as expected:
- If a Web form has either a confidential field or an action URL longer than 256 characters, the field or action URL is truncated in the NetScaler logs.
- With certain SSL transactions, the logs are truncated if either the confidential field or the action URL is longer than 127 characters.

In either of these cases, the application firewall masks a fifteen-character string with the letter "x," instead of the normal eight character string. To ensure that any confidential information is removed, the user must use form field name and action URL expressions that match the first 256, or (in cases where SSL is used) the first 127 characters.

To configure your application firewall to treat a web-form field on a protected web site as confidential, you add that field to the Confidential Fields list. You can enter the field name as a string, or you can enter a PCRE-compatible regular expression specifying one or more fields. You can enable the confidential-field designation when you add the field, or you can modify the designation later.

To add a confidential field by using the command line interface

At the command prompt, type the following commands:

- add appfw confidField <fieldName> <url> [-isRegex (REGEX | NOTREGEX)] [-comment "<string>"] [-state (ENABLED | DISABLED)]
- save ns config

Example

The following example adds all web form fields whose names begin with**Password** to the confidential fields list.

```
add appfw confidField Password "https?://www[.]example[.]com/[^<>]*[^a-z]password[0-9a-z._-]*[.](asp|cgi|htm|html|htp|js|php)" -isRegex REGEX -state ENABLED
save ns config
```

To modify a confidential field by using the command line interface

At the command prompt, type the following commands:

- set appfw confidField <fieldName> <url> [-isRegex (REGEX | NOTREGEX)] [-comment "<string>"] [-state (ENABLED | DISABLED)]
- save ns config

Example

The following example modifies the confidential field designation to add a comment.

```
set appfw confidField Password "https?://www[.]example[.]com/[^<>]*[^a-z]password[0-9a-z._-]*[.](asp|cgi|htm|html|htp|js|php)" -comment "Protect password fields." -isR
save ns config
```

To remove a confidential field by using the command line interface

At the command prompt, type the following commands:

- rm appfw confidField <fieldName> <url>
- save ns config

To configure a confidential field by using the configuration utility

1. Navigate to Security > Application Firewall.
2. In the details pane, under Settings, click Manage Confidential Fields.
3. In the Manage Confidential Fields dialog box, do one of the following:
   - To add a new form field to the list, click Add.
   - To change an existing confidential field designation, select the field, and then click Open.

   The Create Confidential Form Field dialog box or the Configure Confidential Form Field dialog box appears.
   Note: If you select an existing confidential field designation and then click Add, the Create Confidential Form Field dialog box displays the information for that confidential field. You can modify that information to create your new confidential field.
4. In the dialog box, fill out the elements. They are:
   - **Enabled check box.** Select or clear to enable/disable this confidential field designation.
   - **Is form field name a regular expression check box.** Select or clear to enable PCRE-format regular expressions in the form field name.
   - **Field Name.** Enter a literal string or PCRE-format regular expression that either represents a specific field name or that matches multiple fields with names that follow a pattern.
   - **Action URL.** Enter a literal URL or a regular expression that defines one or more URLs of the web page(s) on which the web form(s) that contains the confidential field are located.
   - **Comments.** Enter a comment. Optional.
5. Click Create or OK.
6. To remove a confidential field designation from the confidential fields list, select the confidential field listing you want to remove, then click Remove to remove it, and then click OK to confirm your choice.

7. When you have finished adding, modifying, and removing confidential field designations, click Close.

Examples

Following are some regular expressions that define form field names that you might find useful:

- ^passwd_ (Applies confidential-field status to all field names that begin with the "passwd_" string.)
- ^(([0-9a-zA-Z._-]*||\\x[0-9A-Fa-f][0-9A-Fa-f])+-)?passwd_ (Applies confidential-field status to all field names that begin with the string passwd_, or that contain the string -passwd_ after another string that might contain non-ASCII special characters.)

Following are some regular expressions that define specific URL types that you might find useful. Substitute your own web host(s) and domain(s) for those in the examples.

- If the web form appears on multiple web pages on the web host www.example.com, but all of those web pages are named logon.pl?, you could use the following regular expression:
  https?://www[.]example[.]com/([0-9A-Za-z][0-9A-Za-z_.-]*/)*logon[.]pl\?
- If the web form appears on multiple web pages on the web host www.example-español.com, which contains the n-tilde (ñ) special character, you could use the following regular expression, which represents the n-tilde special character as an encoded UTF-8 string containing C3 B1, the hexadecimal code assigned to that character in the UTF-8 charset:
  https?://www[.]example-espa\xC3\xB1ol[.]com/([0-9A-Za-z][0-9A-Za-z_.-]*/)* logon[.]pl\?
- If the web form containing query.pl appears on multiple web pages on different hosts within the example.com domain, you could use the following regular expression:
  https?://([0-9A-Za-z][0-9A-Za-z_-.]*[.])*example[.]com/([0-9A-Za-z][0-9A-Za-z_-.]*/)*logon[.]pl\?
- If the web form containing query.pl appears on multiple web pages on different hosts in different domains, you could use the following regular expression:
  https?://([0-9A-Za-z][0-9A-Za-z_-.]*[.])*[0-9A-Za-z][0-9A-Za-z_-.]+[.][a-z]{2,6}/([0-9A-Za-z][0-9A-Za-z_-.]*/)*logon[.]pl\?
- If the web form appears on multiple web pages on the web host www.example.com, but all of those web pages are named logon.pl?, you could use the following regular expression:
  https?://www[.]example[.]com/([0-9A-Za-z][0-9A-Za-z_-.]*/)*logon[.]pl\?

Field Types

Jun 12, 2014

A field type is a PCRE-format regular expression that defines a particular data format and minimum/maximum data lengths for a form field in a web form. Field types are used in the Field Formats check.

The application firewall comes with several default field types, which are:

- integer. A string of any length consisting of numbers only, without a decimal point, and with an optional preceding minus sign (-).
- alpha. A string of any length consisting of letters only.
- alphanum. A string of any length consisting of letters and/or numbers.
- nohtml. A string of any length consisting of characters, including punctuation and spaces, that does not contain HTML symbols or queries.
- any. Anything at all.
  Important: Assigning the any field type as the default field type, or to a field, allows active scripts, SQL commands, and other possibly dangerous content to be sent to your protected web sites and applications in that form field. You should use the any type sparingly, if you use it at all.

You can also add your own field types to the Field Types list. For example, you might want to add a field type for a social security number, postal code, or phone number in your country. You might also want to add a field type for a customer identification number or store credit card number.

To add a field type to the Field Types list, you enter the field name as a literal string or PCRE-format regular expression.

To add a field type by using the command line interface

At the command prompt, type the following commands:

- add appfw fieldType <name> <regex> <priority> [-comment "<string>"]
- save ns config

Example

The following example adds a field type named SSN that matches US Social Security numbers to the Field Types list, and sets its priority to 1.

```
add appfw fieldType SSN "^[1-9][0-9]{2,2}-[0-9}{2,2}-[0-9]{4,4}$" 1
save ns config
```

To modify a field type by using the command line interface

At the command prompt, type the following commands:

- set appfw fieldType <name> <regex> <priority> [-comment "<string>"]
- save ns config

Example

The following example modifies the field type to add a comment.

```
set appfw fieldType SSN "^[1-9][0-9]{2,2}-[0-9}{2,2}-[0-9]{4,4}$" 1 -comment "US Social Security Number"
```

save ns config

To remove a field type by using the command line interface

At the command prompt, type the following commands:

- rm appfw fieldType <name>
- save ns config

To configure a field type by using the configuration utility

1. Navigate to Security > Application Firewall.
2. In the details pane, under Settings, click Manage Field Types.
3. In the Manage Field Types dialog box, do one of the following:
   - To add a new field type to the list, click Add.
   - To change an existing field type, select the field type, and then click Open.

   The Create Field Type dialog box or the Configure Field Type dialog box appears.
   Note: If you select an existing field type designation and then click Add, the dialog box displays the information for that field type. You can modify that information to create your new field type.
4. In the dialog box, fill out the elements. They are:
   - Name
   - Regular Expression
   - Priority
   - Comment
5. Click Create or OK.
6. To remove a field type from the Field Types list, select the field type listing you want to remove, then click Remove to remove it, and then click OK to confirm your choice.
7. When you have finished adding, modifying, and removing field types, click Close.

Examples

Following are some regular expressions for field types that you might find useful:

- ^[1-9][0-9]{2,2}-[0-9}{2,2}-[0-9]{4,4}$ U.S. Social Security numbers
- ^[A-C][0-9]{7,7}$ California driver's license numbers.
- ^+[0-9]{1,3} [0-9() -]{1,40}$ International phone numbers with country codes.
- ^[0-9]{5,5}-[0-9]{4,4}$ U.S. ZIP code numbers.
- ^[0-9A-Za-z][0-9A-Za-z.+_-]{0,25}@([0-9A-Za-z][0-9A-Za-z_-]*[.]){1,4}[A-Za-z]{2,6}$ Email addresses.

XML Content Types

Jun 12, 2014

By default, the application firewall treats files that follow certain naming conventions as XML. You can configure the application firewall to examine web content for additional strings or patterns that indicate that those files are XML files. This can ensure that the application firewall recognizes all XML content on your site, even if certain XML content does not follow normal XML naming conventions, ensuring that XML content is subjected to XML security checks.

To configure the XML content types, you add the appropriate patterns to the XML Content Types list. You can enter a content type as a string, or you can enter a PCRE-compatible regular expression specifying one or more strings. You can also modify the existing XML content types patterns.

To add an XML content type pattern by using the command line interface

At the command prompt, type the following commands:

- add appfw XMLContentType <XMLContenttypevalue> [-isRegex (REGEX | NOTREGEX)]
- save ns config

Example

The following example adds the pattern .*/xml to the XML Content Types list and designates it as a regular expression.

```
add appfw XMLContentType ".*/xml" -isRegex REGEX
```

To remove an XML content type pattern by using the command line interface

At the command prompt, type the following commands:

- rm appfw XMLContentType <XMLContenttypevalue>
- save ns config

To configure the XML content type list by using the configuration utility

1. Navigate to Security > Application Firewall.
2. In the details pane, under Settings, click Manage XML Content Types.
3. In the Manage XML Content Types dialog box, do one of the following:
   - To add a new XML content type, click Add.
   - To modify an existing XML content type, select that type and then click Open.

   The Create XML Content Type or Configure XML Content Type dialog box appears.

   Note: If you select an existing XML content type pattern and then click Add, the dialog box displays the information for that XML content type pattern. You can modify that information to create your new XML content type pattern.
4. In the dialog box, fill out the elements. They are:
   - **Is XML content type a regular expression check box.** Select or clear to enable PCRE-format regular expressions in the form field name.
   - **XML Content Type** Enter a literal string or PCRE-format regular expression that matches the XML content type pattern that you want to add.
5. Click Create.
6. To remove an XML content type pattern from the list, select it, then click Remove to remove it, and then click OK to confirm your choice.

7. When you have finished adding and removing XML content type patterns, click Close.

JSON Content Types

Jun 12, 2014

By default, the application firewall treats files with the content type "application/json" as JSON files.The default setting enables the application firewall to recognize JSON content in requests and responses, and to handle that content appropriately.

You can configure the application firewall to examine web content for additional strings or patterns that indicate that those files are JSON files. This can ensure that the application firewall recognizes all JSON content on your site, even if certain JSON content does not follow normal JSON naming conventions, ensuring that JSON content is subjected to JSON security checks.

To configure the JSON content types, you add the appropriate patterns to the JSON Content Types list. You can enter a content type as a string, or you can enter a PCRE-compatible regular expression specifying one or more strings. You can also modify the existing JSON content types patterns.

To add a JSON content type pattern by using the command line interface

At the command prompt, type the following commands:

- add appfw JSONContentType <JSONContenttypevalue> [-isRegex (REGEX | NOTREGEX)]
- save ns config

Example

The following example adds the pattern .*/json to the JSON Content Types list and designates it as a regular expression.

```
add appfw JSONContentType ".*/json" -isRegex REGEX
```

To configure the JSON content type list by using the configuration utility

1. Navigate to Security > Application Firewall.
2. In the details pane, under Settings, click Manage JSON Content Types.
3. In the Manage JSON Content Types dialog box, do one of the following:
   - To add a new JSON content type, click Add.
   - To modify an existing JSON content type, select that type and then click Open.

   The Create JSON Content Type or Configure JSON Content Type dialog box appears.
   Note: If you select an existing JSON content type pattern and then click Add, the dialog box displays the information for that JSON content type pattern. You can modify that information to create your new JSON content type pattern.
4. In the dialog box, fill out the elements. They are:
   - **Is JSON content type a regular expression check box.** Select or clear to enable PCRE-format regular expressions in the form field name.
   - **JSON Content Type** Enter a literal string or PCRE-format regular expression that matches the JSON content type pattern that you want to add.
5. Click Create or OK.
6. To remove a JSON content type pattern from the list, select it, then click Remove to remove it, and then click OK to confirm your choice.
7. When you have finished adding and removing XML content type patterns, click Close.

Logs, Statistics, and Reports

Sep 03, 2013

The information maintained in the logs and statistics, and displayed in the reports, provides important guidance for configuring and maintaining the application firewall.

The Application Firewall Logs

The logs provide information about the requests and responses that the application firewall has observed while protecting your web sites and applications. Most important, it logs each connection that matches a signature or a security check. You can observe the logs to determine which connections are matching a signature or security check. You can then use this information, along with your own knowledge about your protected web sites and applications, to determine whether the connections that each signature or check is matching are valid (false positives). If they are, you can either remove the signature or check from your configuration, or take appropriate measures to mitigate the false positives before you enable blocking for that signature or security check.

NetScaler Format Logs

When configured to use NetScaler format logs, the application firewall produces logs that follow the same format as other NetScaler features. Each log contains the following fields:

- **Timestamp.** The date and time when the connection occurred.
- **Severity.** The severity level of the log.
- **Module.** The NetScaler module that generated the log entry.
- **Event Type.** The type of event, such as signature violation or security check violation.
- **Event ID.** The ID assigned to the event.
- **Client IP.** The IP address of the user whose connection was logged.
- **Transaction ID**. The ID assigned to the transaction that caused the log.
- **Session ID.** The ID assigned to the user session that caused the log.
- **Message.** The log message. Contains information identifying the signature or security check that triggered the log entry.

You can search on any of these fields, or any combination of information from different fields, to select logs to display, limited only by the capabilities of the tools you use to view the logs. You can observe the signatures by using the application firewall wizard to access the NetScaler syslog viewer, or manually by logging onto the NetScaler appliance or NetScaler virtual appliance.

Viewing the Application Firewall Logs

You can view the logs by using the syslog viewer, or by logging onto the NetScaler appliance, opening a Unix shell, and using the Unix text editor of your choice.

- **Viewing by using the syslog viewer.** You invoke the syslog viewer from one of two locations: the Select Signature Actions page or the Select Advanced Actions page in the Application Firewall Wizard. To invoke the syslog viewer for a signature, in the Select Signature Actions pane click the logs link to the right of that signature. To invoke the syslog viewer for a security check, in the Select Advanced Actions page, security checks list, select that security check, and then beneath the list click the Logs button. Either procedure causes the configuration utility to download the current `ns.log` file and then display the entries that are relevant to that signature or security check.

The syslog viewer contains the following elements:

- *Module list box.* The NetScaler module whose logs you want to view. Always set to APPFW for application firewall logs.
- *Event Type list box.* The type of event. For signatures, this is always APPFW_SIGNATURE_MATCH. For security checks, this is the specific security check that you selected.
- *Severity.* It lets you specify only logs of a specific severity level. Leave blank to see all logs.
- *Find Now button.* Search the nslog.file, using the current criteria, and display the logs that match.
- *Clear button.* Resets your settings to the defaults.
- *Logs display window.* Displays the logs that meet the current criteria. Log information is displayed in several columns that correspond to the log fields for the log format that the application firewall is currently configured to maintain, with an additional column, Deploy, to the extreme left. You can sort the display by clicking a column heading. You can create and implement a relaxation for a signature or security check that is blocking legitimate use of a protected web site or web service by selecting a log that shows the unwanted blocking, and then clicking Deploy.
- *Log directory.* The directory where the logs are stored. If you have archived logs stored in a different directory and want to view those, you can click Browse and browse to that directory to display those logs in the Log files list.
- *Log files list.* A list of the log files in the Log directory. To download and uncompress an archived log file, select the file, and then click Download. To refresh the display, click Refresh.
- *Search in list box.* Searches in a particular section of logs when selecting logs to display in the Logs display window. To search something other than the log message, select a different choice.
- *Search string.* Search for the specified string or regular expression to choose the logs to display in the Logs display window. This field is filled out by the application firewall wizard for you with the appropriate value to display the logs relevant to the signature or security check that you selected. You can modify the string to choose logs based on different criteria.
- *Case Sensitive check box.* Select if the Search string is case sensitive.
- *Regular Expression check box.* Select if the Search string is a regular expression.
- *Clear button.* Resets the syslog viewer to its default settings.
- *Go button.* Uses the new search criteria to search the ns.log file and displays the results in the Logs display window.

For more information about the Application Firewall Wizard, see "The Application Firewall Wizard."

- **Viewing from the command line.** Log onto the application firewall appliance, and then type the following command at the NetScaler command prompt:

  shell

  After the Unix shell is displayed, type the following command to navigate to the directory where the logs are stored:

  cd /var/log

  You can use the vi editor, or any Unix text editor or text search tool of your choice to view and filter the logs for specific entries.

  Note: If the text editor or text search tool is not installed by default on the NetScaler appliance, you must first install it before you can use it to view and filter the logs.

The Application Firewall Statistics

When you enable the statistics action for application firewall signatures or security checks, the application firewall maintains information about connections that match that signature or security check. You can view the accumulated statistics information on the Monitoring tab of the main logon page of your application firewall appliance by selecting one of the following choices in the Select Group list box:

- **Application Firewall.** A summary of all statistics information gathered by your application firewall appliance for all profiles.
- **Application Firewall (per profile).** The same information, but displayed per-profile rather than summarized.

You can use this information to monitor how your application firewall is operating and determine whether there is any abnormal activity or abnormal amounts of hits on a signature or security check. If you see such a pattern of abnormal activity, you can check the logs for that signature or security check, to diagnose the issue, and then take corrective action.

The Application Firewall Reports

The application firewall reports provide information about your application firewall configuration and how it is handling traffic for your protected web sites.

The PCI DSS Report

The Payment Card Industry (PCI) Data Security Standard (DSS), version 1.2, consists of twelve security criteria that most credit card companies require businesses who accept online payments via credit and debit cards to meet. These criteria are designed to prevent identity theft, hacking, and other types of fraud. If an internet service provider or online merchant does not meet the PCI DSS criteria, that ISP or merchant risks losing authorization to accept credit card payments through its web site.

ISPs and online merchants prove that they are in compliance with PCI DSS by having an audit conducted by a PCI DSS Qualified Security Assessor (QSA) Company. The PCI DSS report is designed to assist them both before and during the audit. Before the audit, it shows which application firewall settings are relevant to PCI DSS, how they should be configured, and (most important) whether your current application firewall configuration meets the standard. During the audit, the report can be used to demonstrate compliance with relevant PCI DSS criteria.

The PCI DSS report consists of a list of those criteria that are relevant to your application firewall configuration. Under each criterion, it lists your current configuration options, indicates whether your current configuration complies with the PCI DSS criterion, and explains how to configure the application firewall so that your protected web site(s) will be in compliance with that criterion.

The PCI DSS report is located under System > Reports. To generate the report as an Adobe PDF file, click Generate PCI DSS Report. Depending on your browser settings, the report is displayed in the pop-up window or you are prompted to save it to your hard disk.

Note: To view this and other reports, you must have the Adobe Reader program installed on your computer.
The PCI DSS report consists of the following sections:

- **Description.** A description of the PCI DSS Compliance Summary report.
- **Firewall License and Feature Status.** Tells you whether the application firewall is licensed and enabled on your NetScaler appliance.
- **Executive Summary.** A table that lists the PCI DSS criteria and tells you which of those criteria are relevant to the application firewall.
- **Detailed PCI DSS Criteria Information.** For each PCI DSS criterion that is relevant to your application firewall configuration, the PCI DSS report provides a section that contains information about whether your configuration is currently in compliance and, if it is not, how to bring it into compliance.

- **Configuration.** Data for individual profiles, which you access either by clicking Application Firewall Configuration at the top of the report, or directly from the Reports pane. The Application Firewall Configuration report is the same as the PCI DSS report, with the PCI DSS-specific summary omitted, and is described below.

The Application Firewall Configuration Report

The Application Firewall Configuration report is located under System > Reports. To display it, click Generate Application Firewall Configuration Report. Depending on your browser settings, the report is displayed in the pop-up window or you are prompted to save it to your hard disk.

The Application Firewall Configuration report starts with a Summary page, which consists of the following sections:

- **Application Firewall Policies.** A table that lists your current application firewall policies, showing the policy name, the content of the policy, the action (or profile) it is associated with, and global binding information.
- **Application Firewall Profiles.** A table that lists your current application firewall profiles and indicates which policy each profile is associated with. If a profile is not associated with a policy, the table displays INACTIVE in that location.

To download all report pages for all policies, at the top of the Profiles Summary page click Download All Profiles. You display the report page for each individual profile by selecting that profile in the table at the bottom of the screen. The Profile page for an individual profile shows whether each check action is enabled or disabled for each check, and the other configuration settings for the check.

To download a PDF file containing the PCI DSS report page for the current profile, click Download Current Profile at the top of the page. To return to the Profiles Summary page, click Application Firewall Profiles. To go back to the main page, click Home. You can refresh the PCI DSS report at any time by clicking Refresh in the upper right corner of the browser. You should refresh the report if you make changes to your configuration.

Configuring the Application Firewall Logs

Sep 03, 2013

You can configure the application firewall logs by using the configuration utility or the NetScaler command line. You can configure the application firewall to produce logs in either native NetScaler format, or in Common Event Format (CEF).

To configure the Application Firewall logs by using the command line interface

At the command prompt, type the following commands:

- set appfw settings -CEFLogging (**ON** | **OFF**)
- save ns config

Example

The following example configures the application firewall to use CEF logs.

```
set appfw settings -CEFLogging ON
save ns config
```

The following example disables CEF logs and returns the application firewall configuration to using native NetScaler format logs.

```
set appfw settings -CEFLogging OFF
save ns config
```

Appendices

Sep 30, 2013

The following supplemental material provides additional detail about complex or peripheral application firewall tasks.

PCRE Character Encoding Format

Mar 28, 2012

The NetScaler operating system supports direct entry of characters in the printable ASCII character set only—characters with hexadecimal codes between HEX 20 (ASCII 32) and HEX 7E (ASCII 127). To include a character with a code outside that range in your application firewall configuration, you must enter its UTF-8 hexadecimal code as a PCRE regular expression.

A number of character types require encoding using a PCRE regular expression if you include them in your application firewall configuration as a URL, form field name, or Safe Object expression. They include:

- **Upper-ASCII characters.** Characters with encodings from HEX 7F (ASCII 128) to HEX FF (ASCII 255). Depending on the character map used, these encodings can refer to control codes, ASCII characters with accents or other modifications, non-Latin alphabet characters, and symbols not included in the basic ASCII set. These characters can appear in URLs, form field names, and safe object expressions.
- **Double-Byte characters.** Characters with encodings that use two 8-byte words. Double-byte characters are used primarily for representing Chinese, Japanese, and Korean text in electronic format. These characters can appear in URLs, form field names, and safe object expressions.
- **ASCII control characters.** Non-printable characters used to send commands to a printer. All ASCII characters with hexadecimal codes less than HEX 20 (ASCII 32) fall into this category. These characters should never appear in a URL or form field name, however, and would rarely if ever appear in a safe object expression.

The NetScaler appliance does not support the entire UTF-8 character set, but only the characters found in the following eight charsets:

- **English US (ISO-8859-1).** Although the label reads, “English US,” the application firewall supports all characters in the ISO-8859-1 character set, also called the Latin-1 character set. This character set fully represents most modern western European languages and represents all but a few uncommon characters in the rest.
- **Chinese Traditional (Big5).** The application firewall supports all characters in the BIG5 character set, which includes all of the Traditional Chinese characters (ideographs) commonly used in modern Chinese as spoken and written in Hong Kong, Macau, Taiwan, and by many people of Chinese ethnic heritage who live outside of mainland China.
- **Chinese Simplified (GB2312).** The application firewall supports all characters in the GB2312 character set, which includes all of the Simplified Chinese characters (ideographs) commonly used in modern Chinese as spoken and written in mainland China.
- **Japanese (SJIS).** The application firewall supports all characters in the Shift-JIS (SJIS) character set, which includes most characters (ideographs) commonly used in modern Japanese.
- **Japanese (EUC-JP).** The application firewall supports all characters in the EUC-JP character set, which includes all characters (ideographs) commonly used in modern Japanese.
- **Korean (EUC-KR).** The application firewall supports all characters in the EUC-KR character set, which includes all characters (ideographs) commonly used in modern Korean.
- **Turkish (ISO-8859-9).** The application firewall supports all characters in the ISO-8859-9 character set, which includes all letters used in modern Turkish.

- **Unicode (UTF-8).** The application firewall supports certain additional characters in the UTF-8 character set, including those used in modern Russian.

When configuring the application firewall, you enter all non-ASCII characters as PCRE-format regular expressions using the hexadecimal code assigned to that character in the UTF-8 specification. Symbols and characters within the normal ASCII character set, which are assigned single, two-digit codes in that character set, are assigned the same codes in the UTF-8 character set. For example, the exclamation point (!), which is assigned hex code 21 in the ASCII character set, is also hex 21 in the UTF-8 character set. Symbols and characters from another supported character set have a paired set of hexadecimal codes assigned to them in the UTF-8 character set. For example, the letter a with an acute accent (á) is assigned UTF-8 code C3 A1.

The syntax you use to represent these UTF-8 codes in the application firewall configuration is "\xNN" for ASCII characters; "\xNN\xNN" for non-ASCII characters used in English, Russian, and Turkish; and "\xNN\xNN\xNN" for characters used in Chinese, Japanese, and Korean. For example, if you want to represent a ! in an application firewall regular expression as a UTF-8 character, you would type \x21. If you want to include an á, you would type \xC3\xA1.

Note: Normally you do not need to represent ASCII characters in UTF-8 format, but when those characters might confuse a web browser or an underlying operating system, you can use the character's UTF-8 representation to avoid this confusion. For example, if a URL contains a space, you might want to encode the space as \x20 to avoid confusing certain browsers and web server software.
Below are examples of URLs, form field names, and safe object expressions that contain non-ASCII characters that must be entered as PCRE-format regular expressions to be included in the application firewall configuration. Each example shows the actual URL, field name, or expression string first, followed by a PCRE-format regular expression for it.

- A URL containing extended ASCII characters.

  Actual URL: `http://www.josénuñez.com`

  Encoded URL: `^http://www[.]jos\xC3\xA9nu\xC3\xB1ez[.]com$`

- Another URL containing extended ASCII characters.

  Actual URL: `http://www.example.de/trömso.html`

  Encoded URL: `^http://www[.]example[.]de/tr\xC3\xB6mso[.]html$`

- A form field name containing extended ASCII characters.

  Actual Name: `nome_do_usuário`

  Encoded Name: `^nome_do_usu\xC3\xA1rio$`

- A safe object expression containing extended ASCII characters.

  Unencoded Expression `[A-Z]{3,6}¥[1-9][0-9]{6,6}`

  Encoded Expression: `[A-Z]{3,6}\xC2\xA5[1-9][0-9]{6,6}`

You can find a number of tables that include the entire Unicode character set and matching UTF-8 encodings on the Internet. A useful web site that contains this information is located at the following URL:

`http://www.utf8-chartable.de/unicode-utf8-table.pl`

For the characters in the table on this web site to display correctly, you must have an appropriate Unicode font installed on your computer. If you do not, the visual display of the character may be in error. Even if you do not have an appropriate font installed to display a character, however, the description and the UTF-8 and UTF-16 codes on this set of web pages will be correct.

Whitehat WASC Signature Types for WAF Use

May 02, 2013

The Citrix NetScaler Application Firewall accepts and generates blocking rules for all vulnerability types that the Whitehat scanners generate. However, certain vulnerabilities are most applicable to a web application firewall. Following are lists of those vulnerabilities, categorized by whether they are addressed by WASC 1.0, WASC 2.0, or best practices signature types.

WASC 1.0 Signature Types

- HTTP Request Smuggling
- HTTP Response Splitting
- HTTP Response Smuggling
- Null Byte Injection
- Remote File Inclusion
- URL Redirector Abuse

WASC 2.0 Signature Types

- Abuse of Functionality
- Brute Force
- Content Spoofing
- Denial of Service
- Directory Indexing
- Information Leakage
- Insufficient Anti-automation
- Insufficient Authentication
- Insufficient Authorization
- Insufficient Session Expiration
- LDAP Injection
- Session Fixation

Best Practices

- Autocomplete Attribute
- Insufficient Cookie Access Control
- Insufficient Password Strength
- Invalid HTTP Method Usage
- Non-HttpOnly Session Cookie
- Persistent Session Cookie
- Personally Identifiable Information
- Secured Cachable HTTP Messages
- Unsecured Session Cookie

Content Filtering

Sep 03, 2013

Content filtering can do some of the same tasks as the Citrix NetScaler Application Firewall, and is a less CPU-intensive tool. It is limited, however, to examining the header portion of the HTTP request or response and to performing a few simple actions on connections that match. If you have a complex Web site that makes extensive use of scripts and accesses back-end databases, the Application Firewall may be the better tool for protecting that Web site. For more information about the Citrix NetScaler Application Firewall, see the *Citrix Application Firewall Guide* at http://support.citrix.com/article/CTX132360.

Content filtering is based on regular expressions that you can apply to either HTTP requests or HTTP responses. To block requests from a particular site, for example, you could use an expression that compares each request's URL to the URL specified in the expression. The expression is part of a policy, which also specifies an action to be performed on requests or responses that match the expression. For example, an action might drop a request or reset the connection.

Following are some examples of things you can do with content filtering policies:

- Prevent users from accessing certain parts of your Web sites unless they are connecting from authorized locations.
- Prevent inappropriate HTTP headers from being sent to your Web server, possibly breaching security.
- Redirect specified requests to a different server or service.

To configure content filtering, once you have made sure that the feature is enabled, you configure filtering actions for your servers to perform on selected connections (unless the predefined actions are adequate for your purposes). Then you can configure policies to apply the actions to selected connections. Your policies can use predefined expressions, or you can create your own. To activate the policies you configured, you bind them either globally or to specific virtual servers.

To configure content filtering, do the following:

1. Enabling Content Filtering
2. Configuring a Content Filtering Action
3. Configuring a Content Filtering Policy
4. Binding a Content Filtering Policy

Enabling Content Filtering

Jul 22, 2015

By default, content filtering is enabled on NetScaler appliances running the NetScaler operating system 8.0 or above. If you are upgrading an existing appliance from an operating system version earlier than 8.0, you must update the licenses before you can use content filtering, and you may need to enable the content filtering feature itself manually.

At the command prompt, type the following commands to enable content filtering and verify the configuration:

- enable ns feature ContentFiltering
- show ns feature

Example

```
> enable ns feature ContentFiltering
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL               ON
 2)    Surge Protection          SP               OFF
 .
 .
 .
 .
 11)   Http DoS Protection       HDOSP            OFF
 12)   Content Filtering         CF               ON
 .
 .
 23)   HTML Injection            HTMLInjection    ON
 24)   NetScaler Push            push             OFF
Done
```

1. In the navigation pane, expand System, and then select Settings.
2. In the details pane, click Configure basic features in the Modes and Features section.
3. In the Configure Basic Features dialog box, select the Content Filter check box, and then click OK.

Configuring a Content Filtering Action

Jul 22, 2015

After you enable the content filtering feature, you create one or more actions to tell your NetScaler appliance how to handle the connections it receives.

Content filtering supports the following actions for HTTP requests:

Add

Adds the specified HTTP header before sending the request to the Web server.

Reset

Terminates the connection, sending the appropriate termination notice to the user's browser.

Forward

Redirects the request to the designated service.

Drop

Silently deletes the request, without sending a response to the user's browser.

Corrupt

Modifies the designated HTTP header in a manner that prevents it from performing the function it was intended to perform, then sends the request to the server.

Content filtering supports the following actions for HTTP responses:

Add

Adds the specified HTTP header before sending the response to the user's browser.

ErrorCode

Returns the designated HTTP error code to the user's browser.

Corrupt

Modifies the designated HTTP header in a manner that prevents it from performing the function it was intended to perform, then sends the response to the user's browser.

At the command prompt, type the following commands to configure a Content Filtering action and verify the configuration:

- add filter action <name> <qualifier> [<serviceName>] [<value>] [<respCode>] [<page>]
- show filter action <name>

Example

```
> add filter action act_drop Drop
 Done
> show filter action act_drop
1)     Name: act_drop   Filter Type: drop
 Done
```

1. Navigate to Security > Protection Features > Filter.
2. In the details pane, do one of the following:
   - To create a new action, in the Actions tab, click Add.

- To modify an existing action, select the action, and then click Open.

3. In the **Create** Filter Action or Configure Filter Action dialog box, specify values for the parameters:
   - Action Name*—name
   - Qualifier*—qualifier (Determines which of the following parameters you can configure)

     Reset

     Add (HeaderName:Value)

     Corrupt (Header Name)

     Forward (Service Name)

     ErrorCode (Response Code and corresponding Response page)

     Drop

   4. Fill in any other required information. For example, if you are configuring an action to send an HTTP error code, you must choose the appropriate error code from a drop-down list. If necessary, you can then modify the text of the error message, which is displayed beneath the drop-down list.

   Click Create or OK, and then click Close. The Actions list displays the action you configured, and a message in the status bar indicates that your action has been created.

Configuring a Content Filtering Policy

Jul 22, 2015

To implement content filtering, you must configure at least one policy to tell your NetScaler appliance how to distinguish the connections you want to filter. You must first have configured at least one filtering action, because when you configure a policy, you associate it with an action.

Content filtering policies examine a combination of one or more of the following elements to select requests or responses for filtering:

URL

The URL in the HTTP request.

URL query

Only the query portion of the URL, which is the portion after the query (?) symbol.

URL token

Only the tokens in the URL, if any, which are the parts that begin with an ampersand (&) and consist of the token name, followed by an equals sign (=), followed by the token value.

HTTP method

The HTTP method used in the request, which is usually GET or POST, but can be any of the eight defined HTTP methods.

HTTP version

The HTTP version in the request, which is usually HTTP 1.1.

Standard HTTP header

Any of the standard HTTP headers defined in the HTTP 1.1 specification.

Standard HTTP header value

The value portion of the HTTP header, which is the portion after the colon and space (:).

Custom HTTP header

A non-standard HTTP header issued by your Web site or that appears in a user request.

Custom header value

The value portion of the custom HTTP header, which (as with the standard HTTP header) is the portion after the colon and space (:).

Client Source IP

The IP from which the client request was sent.

Content filtering policies use the simpler of two NetScaler expressions languages, called classic expressions. For a complete description of classic expressions, how they work, and how to configure them manually, see "Policies and Expressions."

Note: Users who are not experienced in configuring policies at the NetScaler command line will usually find using the configuration utility considerably easier.

At the command prompt, type the following commands to configure a content filtering policy and verify the configuration:

- add filter policy <name> -rule <expression> (-reqAction <action> | -resAction <string>
- show filter policy <name>

Example

```
> add filter policy cf-pol -rule "REQ.HTTP.URL CONTAINS http://abc.com" -reqaction DROP
 Done
> show filter policy cf-pol
1)     Name: cf-pol    Rule: REQ.HTTP.URL CONTAINS http://abc.com
       Request action: DROP
       Response action:
       Hits: 0
 Done
```

1. Navigate to Security > Protection Features > Filter.
2. Select the **Policies** tab.
3. In the details pane, to create a new policy, click **Add**.
4. If you are creating a new policy, in the Create Filter Policy dialog box, in the ***Filter Name** text box, type a name for your new policy.
5. Select either **Request Action** or **Response Action** to activate the drop-down list to the right of that item.
6. Click the down arrow to the right of the drop-down list and select the action to be performed on the request or response. The default choices are **RESET** and **DROP**. Any other actions you have created will also appear in this list.
   Note: You can also click New to create a new Content Filtering action, or Modify to modify an existing Content Filtering action. You can only modify actions you created; the default actions are read-only.
7. If you want to use a predefined expression (or named expression) to define your policy, choose one from the Named Expressions list.
   1. Click the down arrow to the right of the first **Named Expressions** drop-down list, and choose the category of named expressions that contains the named expression you want to use.
   2. Click the down arrow to the right of the second Named Expressions drop-down list, and choose the named expression you want. As you choose a named expression, the regular expression definition of that named expression appears in the **Preview Expression** pane beneath the **Named Expression** list boxes.
   3. Click Add Expression to add that named expression to the Expression list.

   Note: You should perform either this step or step 7, but not both.
8. If you want to create a new expression to define your policy, use the Expression Editor.
   1. Click the **Add** button. The Add Expression dialog box appears.
   2. In the Add Expression dialog box, choose the type of connection you want to filter. The Flow Type is set to **REQ** by default, which tells the NetScaler appliance to look at incoming connections, or requests. If you want to filter outgoing connections (responses), you click the right arrow beside the drop-down list and choose **RES**.
   3. If the Protocol is not already set to **HTTP**, click the down arrow to the right of the Protocol drop-down list and choose HTTP.
      Note: In the NetScaler classic expressions language, “HTTP” includes HTTPS requests, as well.
   4. Click the down arrow to the right of the Qualifier drop-down list, and then choose a qualifier for your expression. Your choices are:
      METHOD

      The HTTP method used in the request.
      URL

The contents of the URL header.

URLTOKENS

The URL tokens in the HTTP header.

VERSION

The HTTP version of the connection.

HEADER

The header portion of the HTTP request.

URLLEN

The length of the contents of the URL header.

URLQUERY

The query portion of the contents of the URL header.

URLQUERYLEN

The length of the query portion of the URL header.

The contents of the remaining list boxes change to the choices appropriate to the Qualifier you pick. For example, if you choose HEADER, a text field labeled Header Name* appears below the Flow Type list box.

5. Click the down arrow to the right of the Operator drop-down list, and choose an operator for your expression. Your choices will vary depending on the Protocol you chose in the preceding step. The following list includes all of the operators:

   ==

   Matches the following text string exactly.

   !=

   Does not exactly match the following text string.

   >

   Is greater than the following integer.

   CONTAINS

   Contains the following text string.

   CONTENTS

   The contents of the designated header, URL, or URL query.

   EXISTS

   The specified header or query exists.

   NOTCONTAINS

   Does not contain the following text string.

   NOTEXISTS

   The specified header or query does not exist.

6. If the Value text box is visible, type the appropriate string or number. If you are testing a string in any way, type the string into the Value text box. If you are testing an integer in any way, type the integer into the Value text box.
7. If you chose HEADER as the Protocol, type the header you want in the **Header Name*** text box.
8. Click OK to add your expression to the Expressions list.

9. Repeat steps B through H to create any additional expressions you want for your profile.
10. Click Close to close the Expressions Editor.

9. If you created a new expression, in the Expression frame select an option from the Match Any Expression drop-down list. Your choices are:
   - **Match Any Expression.** If a request matches any expression in the Expressions list, the request matches this policy.
   - **Match All Expressions.** If a request matches all expressions in the Expressions list, the request matches this policy. If it does not matchall of them, it does not match this policy.
   - **Tabular Expressions.** Switches the Expressions list to a tabular format with three columns. In the first column you can place a BEGIN [(] operator. The second column contains the expressions you have selected or created. In the third column, you can place any of the other operators in the following list, to create complex policy groups in which each group can be configured for match any expression or match all expressions.
   - **Advanced Free-form.** Advanced Free-Form Switches off the Expressions Editor entirely and modifies the Expressions list into a text area.

     Note: Content Filter uses classic policy infrastructure. RegEx expressions are not valid in classic policies.

     The **AND** [&&] operator tells the appliance to require that are quest match both the current expression and the following expression.

     The **OR** [||] operator tells the appliance to require that arequest match either the current expression or the following expression, or both. Only if the request does not match either expression does it not match the policy.

     The **END** [)] operator tells the appliance that this is the last expression in this expression group or policy.

     Note: The Tabular format allows you to create a complex policy that contains both “Match Any Expression” and “Match All Expressions” on a per-expression basis. You are not limited to just one or the other.

10. Repeat steps 6 through 8 to add any additional expressions you want to the Expressions list. You can mix named expressions and expressions created in the Expressions Editor. To the NetScaler appliance, they are all the same.
11. Click **Create** to create your new policy. Your new policy appears in the Policies pane list.
12. Click Close. To create additional Content Filtering policies, repeat the previous procedure. To remove a Content Filtering policy, select the policy in the Policies tab and click Remove.

Binding a Content Filtering Policy

Oct 31, 2013

You must bind each content filtering policy to put it into effect. You can bind policies globally or to a particular virtual server. Globally bound policies are evaluated each time traffic directed to any virtual server matches the policy. Policies bound to a specific vserver are evaluated only when that vserver receives traffic that matches the policy.

At the command prompt, type the following commands to bind a policy to a virtual server and verify the configuration:

- bind lb vserver <name>@ -policyName <string> -priority <positive_integer>
- show lb vserver <name>

Example

```
> bind lb vserver vs-loadbal -policyName policyTwo -priority 100
 Done
> show lb vserver vs-loadbal
1)      vs-loadbal (10.102.29.20:80) - HTTP      Type: ADDRESS
        State: OUT OF SERVICE
        Last state change was at Wed Aug 19 09:05:47 2009 (+211 ms)
        Time since last state change: 2 days, 00:58:03.260
        Effective State: DOWN
        Client Idle Timeout: 180 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Port Rewrite : DISABLED
        No. of Bound Services :  0 (Total)       0 (Active)
        Configured Method: LEASTCONNECTION
        Mode: IP
        Persistence: NONE
        Vserver IP and Port insertion: OFF
        Push: DISABLED  Push VServer:
        Push Multi Clients: NO
        Push Label Rule: none

 Done
```

At the command prompt, type the following commands to globally bind a policy and verify the configuration:

- bind filter global (<policyName> [-priority <positive_integer>]) [-state (ENABLED | DISABLED)]
- show filter global

Example

```
bind filter global cf-pol -priority 1
Done show filter global
```

1) Policy Name: cf-pol Priority: 1
Done

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane, select the virtual server to which you want to bind the content filtering policy from the list, and click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, select the Policies tab, and then select the check box in the Active column of the filter policy that you want to bind to the virtual server.
4. Click OK. The policies you have bound display a check mark and the word Yes in the Policies Bound column of the Policies tab.

1. Navigate to Security > Protection Features > Filter.
2. In the details pane, in the Policies tab, select the policy that you want to bind, and then click Global Bindings.
3. In the Bind/Unbind Filter Policies dialog box, in the Policy Name drop-down list, select a policy, and then click Add. The policy is added to the Configured list.
   Note: To select multiple policies from the list, press and hold the Ctrl key, then click each policy you want.
4. Click OK, and then click Close. The policies you have bound display a check mark and the word Yes in the Globally Bound column of the Policies tab.

Configuring Content Filtering for a Commonly Used Deployment Scenario

Oct 31, 2013

This example provides instructions for using the configuration utility to implement a content filtering policy in which, if a requested URL contains root.exe or cmd.exe, the content filtering policy filter-CF-nimda is evaluated and the connection is reset.

To configure this content filtering policy, you must do the following:

- Enable content filtering
- Configure content filtering policy
- Bind content filtering policy globally or to a virtual server
- Verify the configuration

Note: Since this example uses a default content filtering action, you do not need to create a separate content filtering action.

1. In the navigation pane, expand System, and click Settings.
2. In the details pane, under Modes & Features, click Change Basic Features.
3. In the Configure Basic Features dialog box, select the Content Filtering check box, and then click OK.
4. In the Enable/Disable feature(s) dialog box, click Yes. A message appears in the status bar, stating that the selected feature is enabled.

1. Navigate to Security > Protection Features > Filter.
2. In the details pane, click Add. The Create Filter Policy dialog box appears.
3. In the Create Filter Policy dialog box, in the Filter Name text box, type the name filter-CF-nimda.
4. Select the Request Action option, and in the drop-down list, select RESET.
5. In the Expression frame, select Match Any Expression from the drop-down list, and then click Add.
6. In the Add Expression dialog box, Expression Type drop-down list, select General.
7. In the Flow Type drop-down list, select REQ.
8. In the Protocol drop-down list, select HTTP.
9. In the Qualifier drop-down list, select URL.
10. In the Operator drop-down list, select CONTAINS.
11. In the Value text box, type cmd.exe, and then click OK. The expression is added in the Expression text box.
12. To create another expression, repeat Steps 7 through 11, but in the Value text box, type root.exe. Then click OK, and finally click Close.
13. Click Create on the Create Filter Policy dialog box. The filter policy filter-CF-nimda appears in the Filter list.
14. Click Close.

1. Navigate to Security > Protection Features > Filter. The Filter page appears in the right pane.
2. In the details pane, Policies tab, select the policy that you want to bind and click Global Bindings. The Bind/Unbind Filter Policies dialog box appears.

3. In the Bind/Unbind Filter Policies dialog box, in the Policy Name drop-down list, select the policy filter-CF-nimda, and click Add. The policy is added to the Configured list.
4. Click OK, and then click Close. The policy you have bound displays a check mark and Yes in the Globally Bound column of the Policies tab.

1. Navigate to Traffic Management > Load Balancing > Virtual Servers.
2. In the details pane virtual servers list, select vserver-CF-1 to which you want to bind the content filtering policy and click Open.
3. In the Configure Virtual Server (Load Balancing) dialog box, select the Policies tab.
4. In the Active column, select the check box for the policy filter-CF-nimda, and then click OK. Your content filtering policy is now active, and should be filtering requests. If it is functioning correctly, the Hits counter is incremented every time there is a request for a URL containing either root.exe or cmd.exe. This allows you to confirm that your content filtering policy is working. The content filtering policy is bound to the virtual server.

At the command prompt, type the following command to verify the content filtering configuration:
show filter policy filter-CF-nimda

Example

```
sh filter policy filter-CF-nimda
    Name: filter-CF-nimda   Rule: REQ.HTTP.URL CONTAINS cmd.exe || REQ.HTTP.URL CONTAINS root.exe
    Request action: RESET
    Response action:
    Hits: 0
Done
```

Note: The Hits counter displays an integer that denotes the number of times the filter-CF-nimda policy is evaluated. In the preceding steps, the Hits counter is set to zero because no requests for a URL containing either cmd.exe or root.exe have been made yet. If you want to see the counter increment in real time, you can simply request a URL that contains either of these strings.

1. Navigate to Security > Protection Features > Filter.
2. In the details pane, select the filter policy filter-CF-nimda. The bottom of the pane should display the following:

   Request Action:

   RESET

   Rule:

   REQ.HTTP.URL CONTAINS cmd.exe || REQ.HTTP.URL CONTAINS root.exe

   Hits:

   0

Troubleshooting

Jul 22, 2013

If the content filtering feature does not work as expected after you have configured it, you can use some common tools to access NetScaler resources and diagnose the problem.

Updated: 2013-07-22

You can use the following tools and resources to troubleshoot most Content Filtering issues on a NetScaler appliance:

- The Wireshark application customized for the NetScaler trace files
- Trace files recorded when accessing the resource
- The configuration files
- The ns.log file
- The iehttpheaders, or a Fiddler trace or a similar utility

Updated: 2013-08-02

To troubleshoot a content filtering issue, proceed as follows:

- Verify that the feature is enabled.
- Verify that the content filtering policy is configured correctly. Pay special attention to the expression that evaluates the incoming requests.
  Note: Most content filtering issues are caused by incorrect configuration, and the error is most often in the policy configuration.
- Check the policy's Hits counter to verify that it is incrementing. If it is not, the policy is not getting evaluated.
- If the policy is getting evaluated and the required filtering is still not performed, you need to look into the policy expressions and action.
- If the policy's expression seems valid, test it by assigning a simple NSTRUE value to see if the evaluation of the expression is creating any issue.
- Reevaluate whether the filtering should be based on the request or the response.
- Verify that the action is configured correctly. For example, if a custom action is used to corrupt a header in the request, verify that the header name in the action is correct. If you are not sure about the header name, start a browser with iehttpheaders or a similar utility, and then verify the headers in the request. When this feature is used, you can use nstrace to find out if appropriate action is performed when the packets leave NetScaler appliance.
- An iehttpheaders or Fiddler trace can help you find header options and names, client-side request headers, and response headers recorded on the client.
- To check the modifications made to the request header, record an nstrace on the NetScaler appliance or a Wireshark trace on the server.
- If none of the above measures resolves the issue, verify that the connection has not become untrackable, which can happen in certain circumstances. If a connection becomes untrackable, the appliance does not perform any application-level processing of the requests. In that event, contact Citrix Technical Support.

HTTP Denial-of-Service Protection

Mar 16, 2012

Internet hackers can bring down a site by sending a surge of GET requests or other HTTP-level requests. HTTP Denial-of-Service (HTTP Dos) Protection provides an effective way to prevent such attacks from being relayed to your protected Web servers. The HTTP DoS feature also ensures that a NetScaler appliance located between the internet cloud and your Web servers is not brought down by an HTTP DoS attack.

Most attackers on the Internet use applications that discard responses to reduce computation costs, and minimize their size to avoid detection. The attackers focus on speed, devising ways to send attack packets, establish connections or send HTTP requests as rapidly as possible.

Real HTTP clients such as Internet Explorer, Firefox, or NetScape browsers can understand HTML Refresh meta tags, Java scripts, and cookies. In standard HTTP the clients have most of these features enabled. However, the dummy clients used in DoS attacks cannot parse the response from the server. If malicious clients attempt to parse and send requests intelligently, it becomes difficult for them to launch the attack aggressively.

When the NetScaler appliance detects an attack, it responds to a percentage of incoming requests with a Java or HTML script containing a simple refresh and cookie. (You configure that percentage by setting the Client Detect Rate parameter.) Real Web browsers and other Web-based client programs can parse this response and then resend a POST request with the cookie. DoS clients drop the NetScaler appliance's response instead of parsing it, and their requests are therefore dropped as well.

Even when a legitimate client responds correctly to the NetScaler appliance's refresh response, the cookie in the client's POST request may become invalid in the following conditions:

- If the original request was made before the NetScaler appliance detected the DoS attack, but the resent request was made after the appliance had come under attack.
- When the client's think time exceeds four minutes, after which the cookie becomes invalid.

Both of these scenarios are rare, but not impossible. In addition, the HTTP DoS protection feature has the following limitations:

- Under an attack, all POST requests are dropped, and an error page with a cookie is sent.
- Under an attack, all embedded objects without a cookie are dropped, and an error page with a cookie is sent.

The HTTP DoS protection feature may affect other NetScaler features. Using DoS protection for a particular content switching policy, however, creates additional overhead because the policy engine must find the policy to be matched. There is some overhead for SSL requests due to SSL decryption of the encrypted data. Because most attacks are not on a secure network, though, the attack is less aggressive.

If you have implemented priority queuing, while it is under attack a NetScaler appliance places requests without proper cookies in a low-priority queue. Although this creates overhead, it protects your Web servers from false clients. HTTP DoS protection typically has minimal effect on throughput, since the test JavaScript is sent for a small percentage of requests only. The latency of requests is increased, because the client must re-issue the request after it receives the JavaScript. These requests are also queued

To implement HTTP DoS protection, you enable the feature and define a policy for applying this feature. Then you configure your services with the settings required for HTTP DoS. You also bind a TCP monitor to each service and bind your

policy to each service to put it into effect.

Layer 3-4 SYN Denial-of-Service Protection

Mar 21, 2012

Any NetScaler appliance with system software version 8.1 or later automatically provides protection against SYN DoS attacks.

To mount such an attack, a hacker initiates a large number of TCP connections but does not respond to the SYN-ACK messages sent by the victimized server. The source IP addresses in the SYN messages received by the server are typically spoofed. Because new SYN messages arrive before the half-open connections initiated by previous SYN messages time out, the number of such connections increases until the server no longer has enough memory available to accept new connections. In extreme cases, the system memory stack can overflow.

A NetScaler appliance defends against SYN flood attacks by using SYN cookies instead of maintaining half-open connections on the system memory stack. The appliance sends a cookie to each client that requests a TCP connection, but it does not maintain the states of half-open connections. Instead, the appliance allocates system memory for a connection only upon receiving the final ACK packet, or, for HTTP traffic, upon receiving an HTTP request. This prevents SYN attacks and allows normal TCP communications with legitimate clients to continue uninterrupted.

SYN DoS protection on the NetScaler appliance ensures the following:

- The memory of the NetScaler is not wasted on false SYN packets. Instead, memory is used to serve legitimate clients.
- Normal TCP communications with legitimate clients continue uninterrupted, even when the Web site is under SYN flood attack.

In addition, because the NetScaler appliance allocates memory for HTTP connection state only after it receives an HTTP request, it protects Web sites from idle connection attacks.

SYN DoS protection on your NetScaler appliance requires no external configuration. It is enabled by default.

Enabling HTTP DoS Protection

Sep 03, 2013

To configure HTTP DoS protection, you must first enable the feature.

At the command prompt, type the following commands to enable HTTP DoS protection and verify the configuration:

- enable ns feature HttpDoSProtection
- show ns feature

Example

```
> enable ns feature HttpDoSProtection
 Done
> show ns feature

      Feature                 Acronym          Status
      -------                 -------          ------
 1)    Web Logging              WL               ON
 2)    Surge Protection         SP               OFF
 .
 .
 .
 10)   Global Server Load Balancing   GSLB            ON
 11)    Http DoS Protection          HDOSP             ON
 12)   Content Filtering         CF               ON
 .
 .
 23)   HTML Injection           HTMLInjection      ON
 24)   NetScaler Push           push             OFF
Done
>
```

1. Navigate to System > Settings.
2. In the details pane, click Configure Advanced Features.
3. In the Configure Advanced Features dialog box, select the HTTP DoS Protection check box.
4. Click OK.

Defining an HTTP DoS Policy

Sep 03, 2013

After you enable HTTP DoS protection, you next create a policy.

Note: Before changing the default setting for Client Detect Rate, see "Tuning the Client Detection/JavaScript Challenge Response Rate."

At the command prompt, type one of the following commands to configure an HTTP DoS policy and verify the configuration:

- add dos policy <name> -qDepth <positive_integer> [-cltDetectRate <positive_integer>]
- set dos policy <name> -qDepth <positive_integer> [-cltDetectRate <positive_integer>]

Example

```
> add dos policy pol-HTTP-DoS  -qDepth 30
 Done
> set dos policy pol-HTTP-DoS  -qDepth 40
 Done
> show dos policy
1)     Policy: pol-HTTP-DoS    QDepth: 40
 Done
>
```

1. Navigate to Security > Protection Features > HTTP DoS.
2. In the details pane, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the policy, and then click Open.
3. In the Create HTTP DoS Policy or Configure HTTP DoS Policy dialog box, specify values for the parameters:
   - Name*—name (You cannot change the name of an existing policy.)
   - QDepth*—qdepth
   - Client Detect Rate—cltDetectRate (Before changing the default setting for cltDetectRate, see "Tuning the Client Detection/JavaScript Challenge Response Rate.")
4. Click OK to create your new policy. The policy that you created appears in the details pane, and the status bar displays a message indicating that the DoS policy is successfully configured.

Configuring an HTTP DoS Service

Sep 03, 2013

After you configure an HTTP DoS policy, you must configure a service for your policy. The service accepts HTTP traffic that is protected by the HTTP DoS policy.

At the command prompt, type one of the following commands to configure an HTTP DoS service and verify the configuration:

- add service <name>@ (<IP>@ | <serverName>@) HTTP <port> [-maxClient <positive_integer>] [-maxReq <positive_integer>] -state ENABLED
- set service <name>@ (<IP>@ | <serverName>@) HTTP <port> [-maxClient <positive_integer>] [-maxReq <positive_integer>] -state ENABLED

Example

```
> add service ser-HTTP-Dos1 10.102.29.40 HTTP 87
 Done
> set service ser-HTTP-Dos1 -maxReq 20
 Done
> show service
1)     srv-http-10 (10.102.29.30:80) - HTTP
       State: DOWN
       Last state change was at Wed Jul  8 07:49:52 2009
       Time since last state change: 34 days, 00:48:18.700
       Server Name: 10.102.29.30
       Server ID : 0   Monitor Threshold : 0
       Max Conn: 0     Max Req: 0      Max Bandwidth: 0 kbits
       Use Source IP: NO
       Client Keepalive(CKA): NO
       Access Down Service: NO
       TCP Buffering(TCPB): NO
       HTTP Compression(CMP): NO
       Idle timeout: Client: 180 sec   Server: 360 sec
       Client IP: DISABLED
       Cacheable: NO
       SC: OFF
       SP: OFF
       Down state flush: ENABLED
        .
        .
        .

5)     ser-HTTP-Dos1 (10.102.29.40:87) - HTTP
       State: DOWN
       Last state change was at Tue Aug 11 08:23:40 2009
       Time since last state change: 0 days, 00:14:30.300
```

Server Name: 10.102.29.40
Server ID : 0 Monitor Threshold : 0
Max Conn: 0 Max Req: 20 Max Bandwidth: 0 kbits
Use Source IP: NO
Client Keepalive(CKA): NO
Access Down Service: NO
TCP Buffering(TCPB): NO
HTTP Compression(CMP): YES
Idle timeout: Client: 180 sec Server: 360 sec
Client IP: DISABLED
Cacheable: NO
SC: OFF
SP: OFF
Down state flush: ENABLED
Done
>

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, do one of the following:
   - To create a new service, click Add.
   - To modify an existing service, select the service, and then click Open.
3. In the Create Server or Configure Server dialog box, specify values for the following parameters, which correspond to the descriptions in "Parameters for configuring an HTTP DoS service" as follows (asterisk indicates a required parameter):
   - Service Name*—name (You cannot change the name of an existing service.)
   - Server*—IP or serverName (Specify one or the other, not both.)
   - Port*—port
4. If the Enable Service check box is not selected, select it.
5. Select the Advanced tab, and select the Override Global check box to enable those choices.
6. Specify values for the following parameters.
   - Max Clients*—maxClient
   - Max Requests*—maxReq
7. Click Create or OK, and then click Close. The service appears in the list of services.

Binding an HTTP DoS Monitor and Policy

Sep 03, 2013

To put HTTP DoS protection into effect after you have configured an HTTP DoS service, you must bind the monitor, and then bind the service to the HTTP DoS policy.

At the command prompt, type the following commands to bind the monitor to the service and verify the configuration:

- bind lb monitor <monitorName> <serviceName>
- show lb monitor

Example

```
> bind lb monitor tcp ser-HTTP-DoS
 Done
> show lb monitor
1)  Name.......:  ping-default  Type......:    PING  State....ENABLED
2)  Name.......:   tcp-default  Type......:    TCP  State....ENABLED
3)  Name.......:        ping  Type......:    PING  State....ENABLED
4)  Name.......:         tcp  Type......:    TCP  State....ENABLED
5)  Name.......:        http  Type......:    HTTP  State....ENABLED
 .
 .
 .
17)  Name.......:     ldns-dns  Type......: LDNS-DNS  State....ENABLED
 Done
```

At the command prompt, type the following commands to bind the policy to the service and verify the configuration:

bind service <serviceName> -policyName <policyname>

Example

```
> bind service ser-HTTP-DoS -policyName pol-HTTP-DoS
 Done
> show service
1)    srv-http-10 (10.102.29.30:80) - HTTP
      State: DOWN
      Last state change was at Wed Jul  8 07:49:52 2009
      Time since last state change: 34 days, 01:24:58.510
      Server Name: 10.102.29.30
      Server ID : 0   Monitor Threshold : 0
      Max Conn: 0    Max Req: 0    Max Bandwidth: 0 kbits
      Use Source IP: NO
      Client Keepalive(CKA): NO
      Access Down Service: NO
      TCP Buffering(TCPB): NO
      HTTP Compression(CMP): NO
```

Idle timeout: Client: 180 sec Server: 360 sec
Client IP: DISABLED
Cacheable: NO
SC: OFF
SP: ON
Down state flush: ENABLED
.
.
.

4) ser-HTTP-Dos (10.102.29.18:88) - HTTP
State: DOWN
Last state change was at Tue Aug 11 08:19:45 2009
Time since last state change: 0 days, 00:55:05.40
Server Name: 10.102.29.18
Server ID : 0 Monitor Threshold : 0
Max Conn: 0 Max Req: 0 Max Bandwidth: 0 kbits
Use Source IP: NO
Client Keepalive(CKA): NO
Access Down Service: NO
TCP Buffering(TCPB): NO
HTTP Compression(CMP): YES
Idle timeout: Client: 180 sec Server: 360 sec
Client IP: DISABLED
Cacheable: NO
SC: OFF
SP: ON
Down state flush: ENABLED

5) ser-HTTP-Dos1 (10.102.29.40:87) - HTTP
State: DOWN
Last state change was at Tue Aug 11 08:23:40 2009
Time since last state change: 0 days, 00:51:10.110
Server Name: 10.102.29.40
Server ID : 0 Monitor Threshold : 0
Max Conn: 0 Max Req: 20 Max Bandwidth: 0 kbits
Use Source IP: NO
Client Keepalive(CKA): NO
Access Down Service: NO
TCP Buffering(TCPB): NO
HTTP Compression(CMP): YES
Idle timeout: Client: 180 sec Server: 360 sec
Client IP: DISABLED
Cacheable: NO
SC: OFF
SP: OFF
Down state flush: ENABLED

Done
>

1. Navigate to Traffic Management > Load Balancing > Services.
2. In the details pane, select the service that you want to bind, and then click Open.
3. In the Configure Service dialog box, select the Monitor tab, click the name of the monitor you want in the Monitors list, and then click Add. The selected monitor is added to the Configured frame.
4. Select the Policies tab, then select the HTTP DoS tab.
5. Select a policy from the Available Policies list, and then click Add. The policy appears in the Configured Policies list.
6. Click OK, and then click Close. A message appears in the status bar, stating that the service has been configured.

Tuning the Client Detection/JavaScript Challenge Response Rate

Sep 03, 2013

After you have enabled and configured HTTP DoS protection, if more than the maximum specified number of clients are waiting in the NetScaler surge queue for the HTTP DoS service, the HTTP DoS protection function is triggered. The default rate of challenged JavaScript responses sent to the client is one percent of the server response rate. The default response rate is inadequate in many real attack scenarios, however, and may need to be tuned.

For example, assume that the Web server is capable of a maximum of 500 responses/sec, but is receiving 10,000 Gets/sec. If 1% of the server responses are sent as JavaScript challenges, responses are reduced to almost none: 5 client (500 *0.01) JavaScript responses, for 10000 waiting client requests. Only about 0.05% of the real clients receive JavaScript challenge responses. However, if the client detection/JavaScript challenge response rate is very high (for example, 10%, generating 1000 challenge JavaScript responses per second), it may saturate the upstream links or harm the upstream network devices. Exercise care when modifying the default **Client Detect Rate** value.

If the configured triggering surge queue depth is, for example, 200, and the surge queue size is toggling between 199 and 200, the NetScaler toggles between the “attack” and “no-attack” modes, which is not desirable. The HTTP DoS feature includes a window mechanism is provided. When the surge queue size reaches the designated queue depth value, triggering “attack” mode, the surge queue size must fall for the NetScaler to enter “no-attack” mode. In the scenario just described, if the value of WINDOW_SIZE is set to 20, the surge queue size must fall below 180 before the NetScaler enters “no-attack” mode. During configuration, you must specify a value more than the WINDOW_SIZE for the **QDepth** parameter when adding a DoS policy or setting a DoS policy.

The triggering surge queue depth should be configured on the basis of previous observations of traffic characteristics. For more information about setting up a correct configuration, see "Guidelines for HTTP DoS Protection Deployment."

Guidelines for HTTP DoS Protection Deployment

Mar 16, 2012

Citrix recommends you to deploy the HTTP DoS protection feature in a tested and planned manner and closely monitor its performance after the initial deployment. Use the following information to fine-tune the deployment of HTTP DoS Protection.

- The maximum number of concurrent connections supported by your servers.
- The average and normal values of the concurrent connections supported by your servers.
- The maximum output rate (responses/sec) that your server can generate.
- The average traffic that your server handles.
- The typical bandwidth of your network.
- The maximum bandwidth available upstream.
- The limits affecting bandwidth (such as external links, a particular router, or other critical devices on the path that may suffer from a traffic surge).
- Whether allowing a greater number of clients to connect is more important than protecting upstream network devices.

To determine the characteristics of a HTTP DoS attack, you should consider the following issues.

- What is the rate of incoming fake requests that you have experienced in the past?
- What types of requests have you received (complete posts, incomplete gets)?
- Did previous attacks saturate your downstream links? If not, what was the bandwidth?
- What types of source IP addresses and source ports did the HTTP requests have (e.g., IP addresses from one subnet, constant IP, ports increasing by one).
- What types of attacks do you expect in future? What type have you seen in the past?
- Any or all information that can help you tune DoS attack protection.

Priority Queuing

Sep 03, 2013

The priority queuing feature lets you filter incoming HTTP traffic on the basis of categories that you create and define, and prioritize those HTTP requests accordingly. Priority queuing directs high-priority requests to the server ahead of low-priority requests, so that users who need resources for important business uses receive expedited access to your protected Web servers.

Note: The priority queuing feature is not supported in NetScaler 9.2 nCore.
To implement priority queuing, you create priority queuing policies that specify a priority, weight, threshold, and implicit action. When an incoming request matches a priority queuing policy, the request is processed as the associated action indicates. For example, you can create a priority queuing policy that places all matching requests above a certain threshold in a surge queue, while giving priority treatment to other requests.

You can bind up to three priority queuing policies to a single load balancing virtual server. The priority levels are:

Level 1

A Level 1 policy processes priority requests.

Level 2

A Level 2 policy processes requests that should receive responses as soon as Level 1 requests have been cleared from the queue.

Level 3

A Level 3 policy processes non-priority requests that receive responses only after requests in the first two queues have been cleared.

You can use weighted queuing to adjust the relative priority of each of these queues. Weights can range from 0 to 101. A weight of 101 tells the NetScaler appliance to clear all requests in that queue before forwarding any requests in the lower-priority queues to the Web server. A weight of 0 tells the appliance to send requests in that queue to the Web server only when there are no requests waiting in any of the other queues.

You must assign a unique name to each priority queuing policy. Policy names can be up to 127 characters. Multiple policies bound to the same load balancing virtual server cannot have the same priority level. No two virtual servers that have one or more common underlying physical services can have priority queuing configured or enabled on both virtual servers simultaneously.

To configure priority queuing the NetScaler, you perform the following steps:

- Enable the load balancing feature
- Define a server and service
- Define a load balancing virtual server
- Bind the service to the load balancing virtual server
- Enable the priority queuing feature
- Create the priority queuing policies
- Bind the priority queuing policies to the load balancing virtual server
- Enable priority queuing on load balancing virtual server

For information about enabling load balancing, creating servers, creating virtual servers and services, and binding these servers and services, see the *Citrix NetScaler Traffic Management Guide* at http://support.citrix.com/article/CTX132359. For

complete information about policies and expressions, see the *Citrix NetScaler Policy Configuration and Reference Guide* at http://support.citrix.com/article/CTX132362.

Enabling Priority Queuing

Sep 03, 2013

To use the priority queuing feature the NetScaler appliance, you must first enable it.

At the command prompt, type the following commands to enable priority queuing and verify the configuration:

- enable ns feature PriorityQueuing
- show ns feature

Example

```
> enable ns feature PriorityQueuing
 Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection          SP              OFF
 3)    Load Balancing           LB              ON
 .
 .
 .
 8)    Priority Queuing          PQ              ON
 .
 23)   HTML Injection           HTMLInjection      ON
 24)   NetScaler Push           push            OFF
 Done
```

1. Navigate to System > Settings.
2. In the details pane, click Configure advanced features.
3. In the Configure Advanced Features dialog box, select the Priority Queuing check box.
4. Click OK.

Configuring a Priority Queuing Policy

Sep 03, 2013

To configure a priority queuing policy, you can use either the configuration utility or the command line.

Note: For more information about using the command line, see "Command Reference."

At the command prompt, type the following command to configure a priority queuing policy and verify the configuration:

add pq policy <policyName> -rule <expression> -priority <positive_integer> [-weight <positive_integer>] [-qDepth <positive_integer> | -polqDepth <positive_integer>]

Example

```
> add pq policy pol_cgibin -rule "URL == '/cgi-bin/'" -priority 1
 Done
> show pq policy pol_cgibin
1)     Policy: pol_cgibin      Rule: URL == '/cgi-bin/'       Priority: 1     Weight: 10
       Hits: 0
 Done
```

1. Navigate to Security > Protection Features > Priority Queuing.
2. In the details pane, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the policy, and then click Open.
3. If you are creating a new policy, in the Create PQ Policy dialog box, in the Name text box, type a name for your new policy.
   The name can consist of from one to 127 letters, numbers and the hyphen and underscore symbol.

   If you are modifying an existing policy, skip this step. You cannot change the name of an existing policy.

4. In the Rule text box, either enter the policy expression directly, or click New to create a policy expression. If you click New, perform the following steps:
   1. In the Create Expression dialog box, click Add.
   2. In the Add Expression dialog box, leave Expression Type set to General, and in the Flow Type drop-down list, select a Flow Type. Your choices are REQ (for requests) and RES (for responses).
   3. In the Protocol drop-down list, select a protocol. If you selected REQ in the previous step, your choices are HTTP (Web-based connections), SSL (secure Web connections), TCP and IP. If you selected RES in the previous step, your choices are HTTP, TCP and IP.
   4. In the Qualifier drop-down list, select a qualifier.
      Your choices depend upon your selections in the previous step. Common choices are HTTP VERSION (the version of the HTTP connection), HTTP HEADER (the specified HTTP header), TCP SOURCEPORT/ DESTPORT (the source or destination port of a TCP connection), and IP SOURCEIP/DESTIP (the source or destination IP of the connection).

      If you choose HTTP HEADER, the Header text box appears beneath the original row of text boxes. You fill in the name of the HTTP header you want.

      For a complete description of the available choices, see "Policies and Expressions."

5. In the Operator drop-down list, select an operator.
   For a complete description of the available choices, see "Policies and Expressions."

6. In the Value text box, type the value you want to test for.
   This may be a text string or a number, depending upon the context. For a complete description of values appropriate to the specific context, see "Policies and Expressions."

7. Click OK. The expression is added in the Expression text box.
8. Click Create. The expression appears in the Rule text box.

5. In the Priority and Weight text boxes, type numeric values, for example, 1 and 30. For more information about Priority and Weight, see "Setting Up Weighted Queuing."
6. Enter a numeric value for either Queue Depth or Policy Queue Depth, for example 234, and click Create.
   - Queue Depth Defines the total number of waiting clients or requests on the virtual server to which the policy is bound.
   - Policy Queue Depth Defines the total number of waiting clients or requests belonging to the policy.

   The policy is created and appears in the Priority Queuing page.
   Note: To create additional priority queuing policies, repeat the procedure in the preceding section, and click Close after you finish.

Binding a Priority Queuing Policy

Sep 03, 2013

After you create a priority queuing policy, you must bind it to the appropriate virtual server to put it into effect.

At the command prompt, type the following commands to bind a policy and verify the configuration:

- bind lb vserver <name> -policyName <policyname>
- show lb vserver <name>

Example

```
> bind lb vserver lbvip -policyname pol_cgibin
 Done
> show lb vserver lbvip
        lbvip (8.7.6.6:80) - HTTP       Type: ADDRESS
        State: DOWN
        Last state change was at Wed Jul 15 05:54:24 2009 (+782 ms)
        Time since last state change: 26 days, 05:44:37.370
        Effective State: DOWN
        Client Idle Timeout: 180 sec
        Down state flush: ENABLED
        Disable Primary Vserver On Down : DISABLED
        Port Rewrite : DISABLED
        No. of Bound Services :  0 (Total)      0 (Active)
        Configured Method: LEASTCONNECTION
        Mode: IP
        Persistence: NONE
        Vserver IP and Port insertion: OFF
        Push: DISABLED  Push VServer:
        Push Multi Clients: NO
        Push Label Rule: none

1)      Policy : ns_cmp_msapp Priority:0

1)      Priority Queuing Policy : pol_cgibin
 Done
>
```

1. In the navigation pane, locate and select the virtual server to which you want to bind the priority queuing policy.
   - To select a load balancing virtual server, expand Traffic Management > Load Balancing > Virtual Servers, then select the load balancing virtual server that you want..
   - To select a content switching virtual server, expand Traffic Management > Content Switching > Virtual Servers, then select the content switching virtual server that you want..
2. In the Configure Virtual Server dialog box, select the Policies tab.

3. Click the double right-arrow (») symbol to display the complete list of policy types, and then select Priority Queuing from the drop-down list.
4. Click Insert Policy.
5. In the Policy Name row, select the policy that you want to bind from the drop-down list.
6. ClickOK to save your changes.

Setting Up Weighted Queuing

Sep 03, 2013

When priority queuing is implemented, lower-priority requests are typically kept on hold while higher-priority requests are served. The lower-priority requests may therefore be delayed if there is a constant flow of higher-priority requests.

To prevent delays for low-priority requests across multiple priority levels, you can configure weighted queuing for serving requests. The default weights for the priorities are:

- Gold - Priority 1 - Weight 3
- Silver - Priority 2 - Weight 2
- Bronze - Priority 3 - Weight 1

You assign the minimum weight, zero (0), to requests that the NetScaler appliance should send to the server only if no requests are stored in any of the other queues. You assign the maximum weight, 101, to requests that the appliance should send to the server immediately, ahead of any requests stored in any of the other queues. Weights between these two set the relative priority of a particular queue in relation to the other queues. Queues with a higher weight are processed first; queues with a lower weight after the others have been processed. To assign the weights, see "Configuring a Priority Queuing Policy."

Note: The weight assigned to a higher-priority queue must be larger than the weight assigned to a lower-priority queue. For example, the weight assigned to The Gold (Priority 1) queue must be greater than the weight assigned to the Silver (Priority 2) queue.

SureConnect

Sep 03, 2013

You can use the SureConnect feature of the Citrix NetScaler appliance to service all incoming connections with either the requested content or a custom Web page that displays information about a delay in the request being serviced.

When servers are overloaded with the requests, the servers might either respond slowly or not at all. The SureConnect feature enables the NetScaler appliance to detect and compensate such conditions by ensuring that every client request gets serviced in some way, such as either a custom Web page or actual content is sent to the client.

SureConnect is activated when the response time or maximum server connections to a client request exceeds a limit that you have set. The SureConnect browser window displays one of the following:

- A progress bar with the amount of time remaining until the requested content will be available.
- Alternate Web content of your choice (alternate page).
- Both a progress bar and alternate page.
- Complete custom content of your choice.

You can configure whether the SureConnect progress bar alone is displayed or both the progress bar and the alternate page are displayed.

When the server becomes responsive again, the original request for content is served. If the user chooses, the alternate content window can remain in focus.

Subsequent requests from the same user within the same session are served immediately. This can be configured using the settings described later in this section.

SureConnect can be activated when a response is delayed, and when the number of user connections to a given URL exceeds a specified threshold.

SureConnect works with all standard browsers, including Microsoft Internet Explorer, Netscape Navigator, and Mozilla Firefox.

SureConnect is advantageous in the following situations:

- **Full server queue**

  The server can respond fast, but there are too many users. This results in the server's queue being full and unable to process additional client requests.

  SureConnect Solution: In this situation, the SureConnect window is displayed, showing the time left until the content will be available. The alternate page is displayed under the progress bar, if an alternate page has been configured.

- **Large response delay**

  The server response is slow. Typically, if a Web server does not respond to a client request quickly, the user will leave the site.

  SureConnect Solution: When the predicted delay reaches a configured time threshold, the SureConnect window displays the progress bar and the optional alternate page in the client browser.

- **Client time-out**

When the client requests content from a very slow Web site, a time-out message displays in the client browser, and the content is not delivered. The user may leave the site.

SureConnect Solution: The appliance stores the request until the server is no longer busy and delivers the requested content to the client.

- **Server experiencing a traffic surge**

  The server typically responds quickly, but the current load of open connections is greater than the server capacity to serve them. Therefore, the server response is delayed.

  SureConnect Solution: A SureConnect window is displayed in the client browser, showing the time left. The alternate page from the server is also displayed if it has been configured.

Installing SureConnect

Mar 21, 2012

SureConnect files must be installed on the alternate content server, which can be the same as the primary server.

On a Windows server, extract the sc_xx.exe file (where xx is the build number), or on a UNIX server, extract the sc_xx.tar file (where xx is the build number).

Note: You must install SureConnect in the default Web root directory.
If the alternate content server is the same as the primary server, place the SureConnect and alternate content files in any directory under the Web root directory. Specify this path when you add a policy to configure SureConnect. By default, SureConnect files are installed in the /Citrix NetScaler appliance directory under the default Web root directory.

If the alternate content server is different than the primary server, the SureConnect and alternate content files must be in a unique directory under the Web root directory. By default, this unique directory is the /Citrix NetScaler system directory. Specify this path when you add a policy to configure SureConnect.

The following files are extracted:

- Alternate content files (progressbar.htm, alternatepage.htm, and barandpage.htm)
- System-Logo.gif
- Customer-Logo.gif
- Sample.gif
- README.txt.

This section describes how to install SureConnect alternate content on a UNIX server. The following are the prerequisites:

- The UNIX server is running the Apache server.
- The shell with the # prompt is in use.
- Apache is installed in the default location.
- The sc_xx.tar file is downloaded from the organization's Web site into the /var/ftp/incoming directory.

To install SureConnect

1. At the command prompt, navigate to the htdocs directory:
   `cd /usr/local/apache/htdocs`
2. Type the following command:
   `tar xvpf/var/ftp/incoming/sc_xx.tar`

The output from the .tar file is displayed. A /Citrix NetScaler system directory is created under the specified path and the SureConnect files are installed.

Updated: 2013-09-03

This section describes how to install SureConnect alternate content on a Windows server. The following are the prerequisites:

- The server is running the Microsoft Internet Information Server.
- The DOS prompt is being used.

- The SureConnect zip (self-extracting) file is downloaded from the organization Web site using FTP into the C:\inetpub\wwwroot directory.

To install SureConnect on Windows

Do one of the following:

- At the command prompt, navigate to the wwwroot directory:
  `cd c:\inetpub\wwwroot`
- Type the name of the executable file:
  `sc_xx.exe`
- Double-click the sc_xx.exe icon from the Microsoft Windows Explorer Web browser, extract from the compressed file into the default path (for example, the c:\inetpub\wwwroot directory).

Output from the zip file is displayed. A /Citrix NetScaler system directory is created under the specified path, and the SureConnect files are installed.

Configuring SureConnect

Sep 03, 2013

The following topics describe how to configure SureConnect for scenarios involving alternate server failure.

- "Configuring the Response for Alternate Server Failure"
- "Configuring the SureConnect Policies"
- "Customizing the Alternate Content File"
- "Configuring SureConnect for Citrix NetScaler Features"

Updated: 2013-09-03

If the alternate server fails, and the primary server cannot immediately deliver the requested content to the client, SureConnect does not display alternate content from the failed alternate server in the client Web browser.

The Citrix NetScaler appliance automatically sends a response to the client browser. You can customize the server response to display information suited to your needs.

The default response is:

Your Request is being processed... Estimated Time: ____ Secs

Customizing the Default Response

The NetScaler appliance automatically sends the response to the client if the alternate server fails, or if the appliance is configured to send the default response.

To customize the default response of the appliance, create a vsr.htm file (a sample is provided in this section) as follows:

- The file can contain any valid HTML statements other than embedded objects.
- The file size cannot exceed 800 bytes.
- The file must reside on the NetScaler appliance. If you have a high availability (HA) setup, the file must reside on the primary and secondary nodes. Any changes made to the file on the primary node must also be applied to the file on the secondary node.
- Put vsr.htm file in the /etc directory.

Change any of the contents between the </HEAD> and </HTML> tags in the vsr.htm file. Following is the sample content from vsr.htm file. The sections that you can edit are in bold text.

```
HTTP/1.1 200 OK
Server: NS_WS3.0
Content-Type: text/html
Cache-control: no-cache
Pragma: no-cache
Set-Cookie: NSC_BPIP=@@SID@@; path=/
<HTML> <HEAD> <META HTTP-EQUIV="Refresh" CONTENT="0">
</HEAD> <font color=blue size=5>Your request is being processed...
```

**
Estimated Delay: @@DELAY@@ Sec ** </HTML>
Note: Include @@DELAY@@ to display the predicted delayed response time in seconds.

SureConnect with In-Memory response (NS action)

Updated: 2013-09-03

When defining the SureConnect policy by using the add sc policy command, you can configure the NetScaler Appliance to serve alternative content to the client.

To enable SureConnect and configure the in-memory response, perform the following tasks:

- Enable the SureConnect feature on the appliance by using the enable feature SC command
- Define the services by using the add service <servicename> <IP address> <servicetype> <port> command. This identifies the original server for which the SureConnect is configured and the types of services.
- Add a SureConnect policy by using the add sc policy command. You can configure a URL-based policy or a rule-based policy. The incoming requests are validated against the URL or rule you specify in the policy.

Note: You can configure the SureConnect feature on a load balancing virtual server. In that case, perform the following additional actions:

- Enable Load Balancing by using the enable feature LB command.
- Enable SureConnect feature on the virtual server by using the set lb vserver <vservername> -sc ON command.
- Bind services to the virtual server by using the bind lb vserver <name> <serviceName> command.
- Bind policies to the virtual server by using the bind lb vserver <name> -policyname <name> command.

The following example illustrates how to configure SureConnect for the load balancing feature so that SureConnect will display alternative content from the NetScaler appliance.

In this example, two physical servers, with IP addresses, 10.101.3.187 and 10.101.3.188 are load balanced by the NetScaler appliance. The appliance has one configured virtual server, vs-NSact, whose IP address is 10.101.3.201. The file that contains the alternative content is vsr.htm. It is copied from the file system into system memory. Services are loaded until the SureConnect policy triggers, and the appliance supplies the alternate content.

```
enable feature SC LB
add service psvc1 10.101.3.187 http 80
add service psvc2 10.101.3.188 http 80
add lb vserver vs-NSact HTTP 10.101.3.201 80
bind lb vserver vs-NSact psvc1
bind lb vserver vs-NSact psvc2
add sc policy policyNS -url /cgi-bin/*.cgi -delay 400000
-action NS
set sc parameter -vsr /nsconfig/ssl/vsr.htm
bind lb vserver vs-NSact -policyName policyNS
set lb vserver vs-NSact -sc ON
save config
```

Table 1. Parameter values used in this example

Service	
Name	psvc1, psvc2

Server	10.101.3.187, 10.101.3.188
Protocol	HTTP
Port	80
Load Balancing Virtual Server	
Name	vs-NSact
IP Address	10.101.3.201
Protocol	HTTP
Port	80
SureConnect Policy	
Name	policyNS
URL	/cgi-bin/*.cgi
Delay(microseconds)	400000
SC Parameter	
VSR File Name	vsr.htm

1. In the In the navigation pane, navigate to System > Settings. In the Modes and Features pane, perform the following actions:
   1. Click Configure Basic Features, select Load Balancing, and Click Go.
   2. Click Configure Advanced Features, select SureConnect, and Click Go.
2. In the navigation pane, navigate to Security > Protection Features > SureConnect. In the details pane, click Parameters. In the Configure SureConnect Parameters window, browse and select the VSR filename.
3. Navigate to Traffic Management > Load Balancing > Services. In the details pane, click Add. In the **Create Services** window, enter the paramter values as shown in Table 5-1, and click **OK**.
4. Navigate to Traffic Management > Load Balancing > Virtual servers. In the details pane, click Add. In the Create Virtual Server (Load Balancing) dialog box, enter the values shown in Table 5.1 for the Load Balancing Virtual Server parameters and click OK.

5. In the navigation pane, navigate to Traffic Management > Load Balancing > Virtual servers. Select the virtual server vs-NSact and click Open in the details pane. The Configure Virtual system (Load Balancing) dialog box, displays the list of configured services. Select services psvc1 and psvc2 and click OK.
6. In the navigation pane, expand Security > Protection Features > SureConnect. In the details pane, click Add. Create the policy with the values as given in the parameters table.
7. In the navigation pane, navigate to Traffic Management > Load Balancing > Virtual servers. Select the virtual server vs-NSact and click Open in the details pane. In the Configure Virtual system (Load Balancing) dialog box, click the Policies tab. Click >> to expand the features. Select **SureConnect**. When the list of SureConnect polices appear, select policyNS and click OK.
8. In the navigation pane, navigate to Traffic Management > Load Balancing > Virtual servers. Select the virtual server vs-NSact and click Open in the details pane. In the Configure Virtual system (Load Balancing) dialog box, on the Advanced tab, select SC and click OK.

You can configure the following SureConnect policies. The NetScaler appliance matches incoming requests in the order the policies are configured:

- Exact URL-based policies
- Wildcard rule-based policies

Configuring Exact URL Based Policies

Updated: 2013-09-03

When you configure an exact URL based policy, the NetScaler appliance matches the incoming request against the URL that has been configured in the policy. URL based policies take precedence over rule based policies.

At the command prompt, type:

add sc policy <name> [-url <URL> | -rule <expression>] [-delay <usecs>] [-maxConn <positive_integer>] [-action (ACS <altContentSvcName> <altContentPath>) | NS | NOACTION)]

1. Navigate to Security > Protection Features > SureConnect.
2. In the details pane, click Add.
3. In the Create SureConnect Policy dialog box, set the following parameters:
   - Name*
   - URL (Make sure that the URL check box is selected)
   - Value*
   - Delay (microseconds)*
   - Maximum Client Connections
   - Action (Select from the Choose Action list.)
   - Alternate Service Name (if you select ACS as the Action)
   - Alternate Content Path (if you select ACS as the Action)

   *A required parameter
4. Click Create, and click Close. The URL based policy appears in the right pane, and a message displays in the status bar that the policy is successfully configured.

Configuring Wildcard Rule-Based Policies

Updated: 2013-09-03

SureConnect matches the incoming requests to a defined rule, if you configure a rule-based policy.

1. Create the expression(s).
   Use the add expression command to create each expression.
2. Create the rule(s).

   Use the add sc policy command with the -rule expression_logic argument to specify the rule(s). In the -rule expression_logic argument, refer to the expression(s) you created in step 1.

   Repeat this command to create and name each rule.

The following example creates a rule “rule = = /*.cgi”:

```
add vserver vs-lb http 1.1.1.1 80
add expression expr1 url == /cgi-bin/*.cgi
add expression expr2 url == /index.html
add sc policy surecpolicy1 -rule (expr1||expr2) -delay 1000000 -action NS
bind lb vserver vs-lb -policyName surecpolicy1
```

To complete the SureConnect configuration, you will need to enter additional commands, beyond those shown in the example.

1. Navigate to Security > Protection Features > SureConnect.
2. In the details pane, click Add.
3. In the Create SureConnect Policy dialog box, in the Name text box, type the name of the policy.
4. Under What to Monitor, click Expression, and then click Configure.
5. In the Create Expression dialog box, click Add.
6. In the Add Expression dialog box, enter an expression. For example, you can select an Expression Type of General, a Flow Type of REQ, a Protocol of HTTP, a Qualifier of URLQUERY, an Operator of CONTAINS, and in the Value text box, type AA. For more information about expressions, see "Policies and Expressions."
7. Click OK, and click Close.
8. In the Create Expression dialog box, click Create.

Examples of wildcard rules:

“/sports/*” matches all URLs under /sports

“/sports*” matches all URLs whose prefix matches “/sports”, starting at the beginning of the URL.

“/*.jsp” matches all URLs whose file extension is “jsp”

When configuring rule-based policies, first add the more specific rule-based policies, before adding more generic rules (for example, add /cgi-bin/sports*.cgi before adding /cgi-bin/*.cgi).

Displaying the Configured SureConnect Policy

To view the SureConnect policy that you have configured, at the NetScaler command prompt, enter the show sc policy

command.

When SureConnect activates, it can display alternate content from one of the following files that you have configured:

- **progressbar.htm.** Displays the progress information.
- **alternatepage.htm.** Displays an alternate page.
- **barandpage.htm.** Displays both the progress information and an alternate page.

The alternate content files are JavaScript files. During SureConnect installation, these files are copied onto the server that contains the alternate content. These files can contain alternate content (including an alternate page) or references to other files that contain the alternate content.

This section describes the changes you can make to the alternate content file provided by the appliance.

```
//**** DEFINE YOUR VALUES HERE ****
var alt_url = "/Citrix NetScaler system /sample.gif";
var alt_url = "http://www.DomainName.com";
var Citrix NetScaler system _logo = "netscaler_logo.gif";
var our_logo = "netscaler_logo.gif";
var height = 450;
var width = 550;
var top = 200;
var left = 200;
var popunder = "no"; //specify yes for pop-under & no for pop-up
var shift_focus = "yes" //if you want to send pop-up to background on getting primary content else specify no
//**** YOUR DEFINITIONS ENDS HERE ****
```

You can make these changes:

- **var alt_url.** Specify the URL for the alternate content if a file provides the alternate content. For example:
  var alt_url = “/Citrix NetScaler system/sports.htm”
  Note: The alternate content file must be present in the /Citrix NetScaler system directory under the documents root of the Web server.
- **var our_logo.** Specify the image file of your organization logo.
- **var height.** Specify the height of the SureConnect window.
- **var width.** Specify the width of the SureConnect window.
- **var top and var left.** Specify the position of the SureConnect window.
- **var popunder.** Specifies the position of the alternate content window. Specify the value as NO to place the alternate content window above the original window. Specify the value as YES to place the alternate content window beneath the original window.
- **var shift_focus.** Specify the focus of the alternate content window. YES places the pop-up window in the background when getting the primary content. NO always keeps the pop-up window in focus, even when getting the primary content.

Note: For more information, see the README.txt file provided by the appliance with other alternate content files.

Updated: 2013-09-03

This section describes how SureConnect works in combination with the load balancing, content switching, cache redirection, and high availability features of the NetScaler appliance.

Configuring SureConnect for Load Balancing

You can use SureConnect in environments where the primary servers use the load balancing feature, with or without alternate servers. If the load balancing virtual server configured for SureConnect fails, the backup virtual server (if there is one) handles the traffic. Backup virtual servers do not support SureConnect policies.

Note: For information about load balancing, see "Load Balancing."

Configuring SureConnect for Cache Redirection

You can use SureConnect in environments where cache redirection is configured. The primary server is a load balancing virtual server bound to the cache redirection virtual server. Regardless of any rules configured for the cache redirection feature:

- You can configure any URL for SureConnect.
- Once SureConnect is activated for a client, requests from the client are always sent to the origin server.

Configuring SureConnect for High Availability

SureConnect is compatible with NetScaler appliances operating in high availability mode.

Note: If the optional vsr.htm file is used, it must be present in both nodes (primary and secondary) and must use the same name and directory.

Activating SureConnect

Sep 03, 2013

You can set the Citrix NetScaler appliance to activate SureConnect if either of two criteria match. Both criteria are arguments to the add sc policy command, as described here:

- -delay <microseconds>

  The first time the client requests the URL, the appliance records how long the server takes to respond. The appliance will not activate SureConnect until the second time the URL is requested. The first and second requests may be from the same or different clients.

  If you set -delay argument, SureConnect will be activated the second time the delay reaches the threshold you set.

- -maxConn <positive_integer>

  When the appliance receives a request, it checks the number of connections to the server for the configured URL. SureConnect is activated if the number of connections is greater than or equal to the value that you set for the -maxConn argument.

  If you will be providing alternate content to be displayed in the client's Web browser, you should configure the -action argument of the add sc policy command. This specifies for the NetScaler appliance whether the alternate content is coming from a dedicated alternate server (-action ACS) or the appliance (-action NS).

  When SureConnect is activated by the –maxConn argument, the SureConnect window and progress bar are displayed in the client's browser (with an alternate page, if configured).

SureConnect Environments

Sep 03, 2013

The following topics describe SureConnect environments.

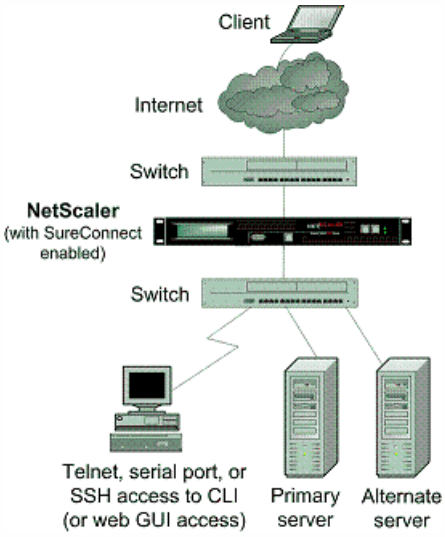
- "Primary and Alternate Servers"
- "Configuration Checklist"
- "Example Configurations"

The SureConnect environment uses a dedicated server to provide alternate content when the requested content is not available. The alternate content may include an alternate page, plus optional components such as frame set, organization logo, and so on. The alternate and primary servers can be the same server.

You can configure SureConnect to display a progress bar when the requested content is not available (or the progress bar and an alternate page).

The following figure illustrates the SureConnect environment.

Figure 1. SureConnect - Primary and Alternate Servers



Updated: 2013-09-03

Complete the following checklist before you start configuration:

Table 1. Configuration Checklist

▫	The same builds are running for the appliance and for the SureConnect files as suggested by appliance staff. Appliance Build Number: _____ SureConnect (sc_xx.exe) Build Number: _____

▫	The latest SureConnect files (style files) are extracted to: • All primary servers (required for NS action). • The alternate content server (required for ACS action).
▫	All customizations to the latest style and vsr.htm files are applied.
▫	The alternate content server is accessible from the Internet (required for ACS action).
▫	If the -redirectURL URL argument of the add vserver CLI command needs to be specified: • The URL is up and running. • This URL is not on the configured servers. • This URL does not match any content in the vserver (that is, do not redirect a missing URL to itself). Redirecting a missing URL to itself can send some browsers into an infinite loop.
▫	All URLs to be configured for SureConnect are top-level URLs only. (Only the URLs that occupy the whole window or frame can be configured, not the embedded objects).

Following are the steps to configure SureConnect in a setup with a primary server and a dedicated alternate server:

- Enable the SureConnect feature
- Add the SureConnect policy
- Bind the SureConnect policy

You can optionally configure the following:

- Redirect the client to another URL if the primary server fails, or send a customized response to the client if the alternate server fails.
- If the servers do not provide alternate content, send a default or customized response.

To redirect the client to another URL

1. Enable the SureConnect feature.
2. Define the primary server and its service.

   You must identify the original server for which SureConnect support is being configured. At the NetScaler command prompt, type the following command:

   ```
   add service <serviceName> <IP> HTTP <port>
   ```

   where <serviceName> assigns a name for the service; <IP> is the server's IP address; and <port> is the port number that the service will use.

   Repeat use of the add service CLI command for each service that is to be added.

   You can also configure SureConnect on a load balancing virtual server. At the NetScaler command prompt, type the following command:

   ```
   add vserver <name> HTTP <IP> <port>
   ```

3. Define and bind the SureConnect policy as follows. If you are configuring a rule-based policy, perform this step as described in "Configuring Wildcard Rule-Based Policies." To configure a URL-based policy, at the NetScaler command prompt, type the following command:

   add sc policy <name> [-url <URL>] [-delay <microsec>] [-maxConn <positiveInteger>]

   For a detailed description of the add sc policy command, see "Command Reference."

   To bind the SureConnect policy, at the NetScaler command prompt, type the following command:

   bind service <serviceName> -policyname <string>
   where <serviceName> is the name of the service defined in step 2, and <string> is the name of the SureConnect policy.

   Repeat the bind service command for each policy created.

   You must include the alternate content page in the altContSvcName argument, and in the altContPath argument of the add sc policy command.

   In the following example, the name of the alternate content file is /Citrix NetScaler system /barandpage.htm, and this file resides in svc2.

4. To save the configuration, at the NetScaler command prompt, type the following command:
   save config

Updated: 2013-09-03

The following examples illustrate various SureConnect configurations.

The examples assume that monitoring of physical services is enabled. If the alternate system is down, SureConnect will deliver the alternate content from the system itself.

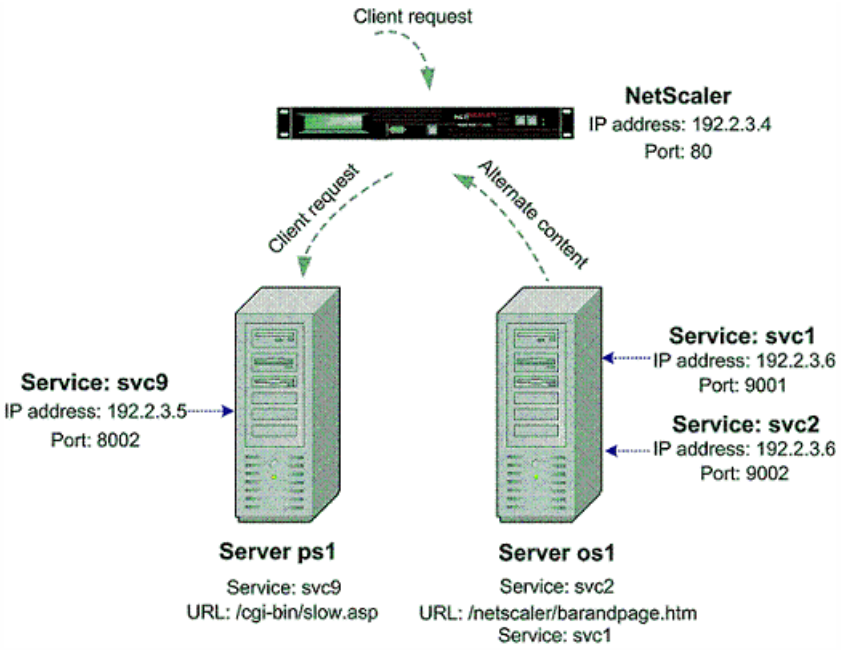
Example 1 - SureConnect Progress Bar and Alternate Page

You can configure SureConnect to display both the progress bar and an alternate page to the user.

To bind a SureConnect policy to a load balancing virtual server, at the command prompt, type the following commands:

bind lb vserver <virtualServerName> -policyName <string>
where <virutalServerName> is the name of the load balancing virtual server defined in step 2 of the configuration process, and <string> is the name of the SureConnect policy defined in step 3.

Figure 2. SureConnect Configuration - Example 1



At the NetScaler command prompt, type the following commands:

```
enable feature SC
show ns info
add service svc2 192.2.3.6 HTTP 9002
show server
show service svc2
add service svc9 192.2.3.5 HTTP 8002
add sc policy policy8 -url /cgi-bin/slow.asp
-delay 3000000 -action ACS svc2 /NetScaler 9000 system barandpage.htm
bind service svc9 -policyname policy8
set service svc9 -sc ON
save config
```

After you configure SureConnect, you can enter commands that show information to verify what you have configured.

Example 2 - SureConnect Progress Bar Only

In this example, SureConnect will display only the progress bar. The server orgsrvr with IP address 10.101.8.187 has service orgsvc. This server is connected to the appliance. The service is bound to the appliance. The progressbar.htm file specifies that only the progress bar will be displayed.

At the NetScaler command prompt, type the following commands:

```
enable feature SC
add service orgsvc 10.101.3.187 HTTP 80
add sc policy policy9 -url /cgi-bin/slow.asp
-delay 4000000 -action ACS orgsvc /NetScaler 9000 system / progressbar.htm
bind service orgsvc -policyname policy9
set service orgsvc -sc ON
save config
```

Example 3 - SureConnect with Load Balancing

This example illustrates how to configure the load balancing feature so that SureConnect will display alternate contents from the primary server. For information about load balancing, see "Load Balancing."

In this example, two physical servers with IP 10.101.3.187 and 10.101.3.188 are being load balanced by the appliance. The name and location of the alternate page file is specified in the file alternatepage.htm, which resides on both servers.

The appliance has one configured virtual server address: 10.101.3.201. At the NetScaler command prompt, type the following commands:

```
enable feature SC LB
add service psvc1 10.101.3.187 HTTP 80
add service psvc2 10.101.3.188 HTTP 80
add vserver vs-SureC HTTP 10.101.3.201 80
bind lb vserver vs-SureC psvc1
bind lb vserver vs-SureC psvc2
add sc policy policy9 -url /cgi-bin/slow.asp -delay 4000000
-action ACS vs-SureC /NetScaler system /alternatepage.htm
bind lb vserver vs-SureC -policyName policy9
set lb vserver vs-SureC -sc ON
save config
```

Example 4 - SureConnect with Load Balancing (ACS Action)

This example illustrates how to configure the NetScaler appliance load balancing feature so that SureConnect will display alternate content from the alternate server. For information about load balancing, see "Load Balancing."

In this case, there are two physical servers, IP 10.101.3.187 and 10.101.3.188. Both are being load balanced by the appliance.

The name and location of the alternate page file are specified in file barandpage.htm, which resides on a third server not being load balanced.

The third server's IP address is 10.101.3.189. Because barandpage.htm is specified, the progress bar and alternate page will both be displayed.

The appliance has one configured virtual server "vsvr" whose IP address (Virtual Server) is 10.101.3.200.

At the NetScaler command prompt, type the following commands:

```
enable feature SC LB
add service psvc1 10.101.3.187 HTTP 80
add service psvc2 10.101.3.188 HTTP 80
add service alt-cont-svc 10.101.3.189 HTTP 80
add vserver vsvr HTTP 10.101.3.200 80
bind lb vserver vsvr psvc1
bind lb vserver vsvr psvc2
add sc policy policy10 -url /cgi-bin/slow.asp
-delay 4000000 -action ACS alt-cont-svc
```

/NetScaler 9000 system /barandpage.htm
bind lb vserver vsvr -policyName policy10
set lb vserver vsvr -sc ON
save config

Example 5 - SureConnect with Content Switching

This example illustrates how to configure SureConnect where the NetScaler content switching and load balancing features are being used. SureConnect is configured on a load balancing virtual server bound to a content switching virtual server.

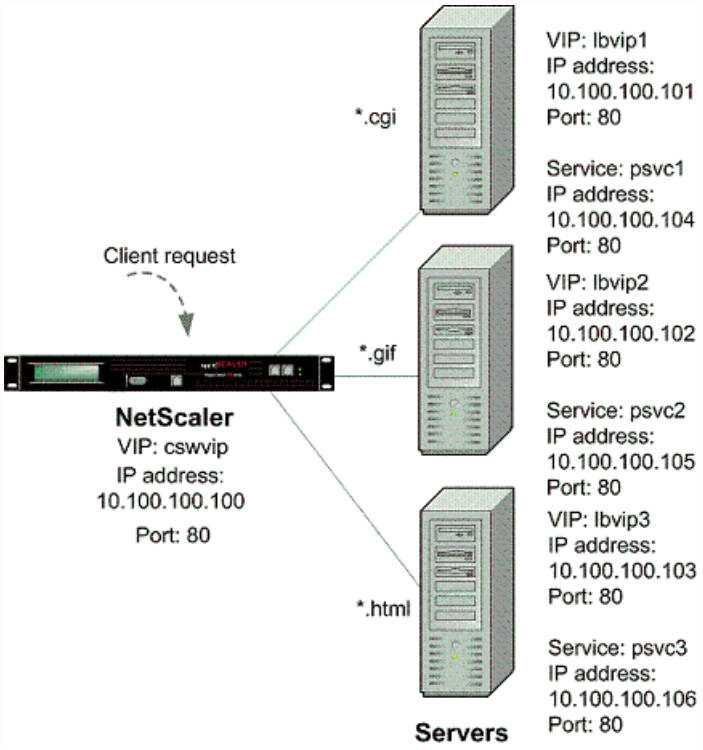
The alternate content is distributed under the content switching virtual server according to the content switching rules. For more information about load balancing and content switching, see "Load Balancing" and "Content Switching."

In this case, three physical services with IP addresses 10.100.100.104, 10.100.100.105, and 10.100.100.106 are bound to three load balancing virtual servers with IP addresses 10.100.100.101, 10.100.100.102, and 10.100.100.103. These three load balancing virtual servers are bound to a content switching virtual server with IP address 10.100.100.100.

In this setup, lbvip1 contains .cgi content, lbvip2 contains .gif content, and .lbvip3 contains .html content.

The name and location of the alternate page file is specified in the file alternatepage.htm, which resides on lbvip3. The embedded objects in this file must be distributed according to the content switching rules (any embedded gif will reside on lbvip2, any embedded htm will reside on lbvip3, and so on).

Figure 3. SureConnect Configuration - Example 5



At the NetScaler command prompt, type the following commands:

enable feature CS LB SC
add vserver cswvip HTTP 10.100.100.100 80 -type CONTENT
add vserver lbvip1 HTTP 10.100.100.101 80 -type ADDRESS

```
add vserver lbvip2 HTTP 10.100.100.102 80 -type ADDRESS
add vserver lbvip3 HTTP 10.100.100.103 80 -type ADDRESS
add service psvc1 10.100.100.104 HTTP 80
add service psvc2 10.100.100.105 HTTP 80
add service psvc3 10.100.100.106 HTTP 80
bind lb vserver lbvip1 psvc1
bind lb vserver lbvip2 psvc2
bind lb vserver lbvip3 psvc3
add cs policy CSWpolicy1 -url /*.cgi
bind cs vserver cswvip lbvip1 -policyName CSWpolicy1
add cs policy CSWpolicy2 -url /*.gif
bind cs vserver cswvip lbvip2 -policyName CSWpolicy2
add cs policy CSWpolicy3 -url /*.htm
bind cs vserver cswvip lbvip3 -policyName CSWpolicy3
add sc policy SCpol -url /cgi-bin/delay.cgi -delay 4000000 -action ACS cswvip /alternatepage.htm
bind lb vserver lbvip1 -policyName SCpol
set lb vserver lbvip1 -sc ON
save config
```

Surge Protection

Mar 21, 2012

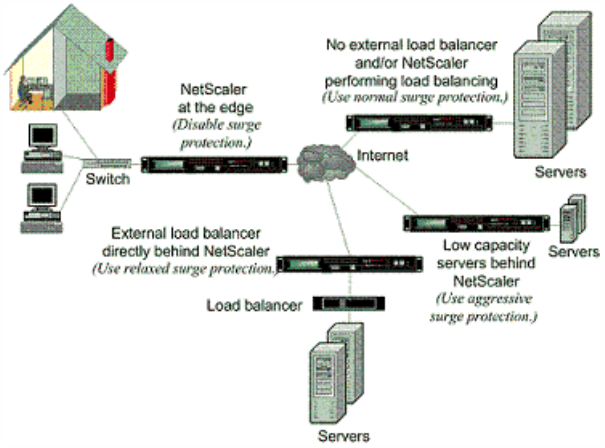
When a surge in client requests overloads a server, server response becomes slow, and the server is unable to respond to new requests. The Surge Protection feature ensures that connections to the server occur at a rate that the server can handle. The response rate depends on how surge protection is configured. The NetScaler appliance also tracks the number of connections to the server, and uses that information to adjust the rate at which it opens new server connections.

Surge protection is enabled by default. If you do not want to use surge protection, as will be the case with some special configurations, you must disable it.

The default surge protection settings are sufficient for most uses, but you can configure surge protection to tune it for your needs. First, you can set the throttle value to tell it how aggressively to manage connection attempts. Second you can set the base threshold value to control the maximum number of concurrent connections that the NetScaler appliance will allow before triggering surge protection. (The default base threshold value is set by the throttle value, but after setting the throttle value you can change it to any number you want.)

The following figure illustrates how surge protection is configured to handle traffic to a Web site.

Figure 1. A Functional Illustration of NetScaler Surge Protection



Note: If the NetScaler appliance is installed at the edge of the network, where it interacts with network devices on the client side of the Internet, the surge protection feature must be disabled. Surge protection must also be disabled if you enable USIP (Using Source IP) mode on your appliance.

The following example and illustration show the request and response rates for two cases. In one case, surge protection is disabled, and in the other it is enabled.

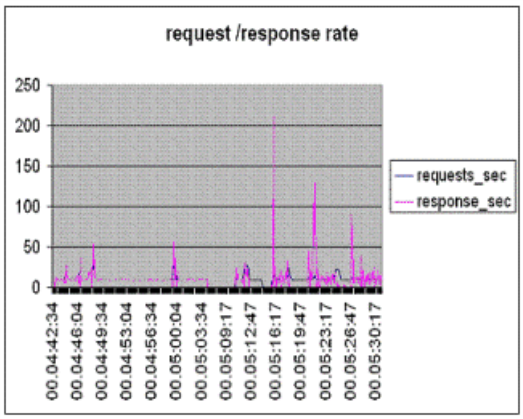
When surge protection is disabled and a surge in requests occurs, the server accepts as many requests as it can process concurrently, and then begins to drop requests. As the server becomes more overloaded, it goes down and the response rate is reduced to zero. When the server recovers from the crash, usually several minutes later, it sends resets for all pending requests, which is abnormal behavior, and also responds to new requests with resets. The process repeats for each surge in requests. Therefore, a server that is under DDoS attack and receives multiple surges of requests can become unavailable to legitimate users.

When surge protection is enabled and a surge in requests occurs, surge protection manages the rate of requests to the server, sending requests to the server only as fast as the server can handle those requests. This enables the server to respond to each request correctly in the order it was received. When the surge is over, the backlogged requests are cleared

as fast as the server can handle them, until the request rate matches the response rate.

The following figure compares the request and response scenarios when surge protection is enabled to that when it is disabled.

Figure 2. Request/Response Rate with and without Surge Protection



Disabling and Reenabling Surge Protection

Sep 03, 2013

The surge protection feature is enabled by default. When surge protection is enabled, it is active for any service that you add.

At the command prompt, type one of the following sets of commands to disable or reenable surge protection and verify the configuration:

- disable ns feature SurgeProtection
- show ns feature

- enable ns feature SurgeProtection
- show ns feature

Example

```
disable ns feature SurgeProtection
 Done show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection         SP              OFF
 .
 .
 .
 23)   HTML Injection           HTMLInjection     ON
 24)   NetScaler Push           push            OFF
Done
enable ns feature SurgeProtection
Done
> show ns feature

      Feature                Acronym          Status
      -------                -------          ------
 1)    Web Logging              WL              ON
 2)    Surge Protection         SP              ON
 .
 .
 .

 23)   HTML Injection           HTMLInjection     ON
 24)   NetScaler Push           push            OFF
Done
>
```

1. In the navigation pane, expand System, and then select Settings.
2. In the details pane, click Change Advanced Features.
3. In the Configure Advanced Features dialog box, clear the selection from the Surge Protection check box to disable the surge protection feature, or select the check box to enable the feature.
4. Click OK.
5. In the Enable/Disable Feature(s) dialog box, click Yes. A message appears in the status bar, stating that the feature has been enabled or disabled.

1. Navigate to Traffic Management > Load Balancing > Services. The list of configured services is displayed in the details pane.
2. In the details pane, select the service for which you want to disable or reenable the surge protection feature, and then click Open.
3. In the Configure Service dialog box, click the Advanced tab and scroll down.
4. In the Others frame, clear the selection from the Surge Protection check box to disable the surge protection feature, or select the check box to enable the feature.
5. Click OK. A message appears in the status bar, stating that the feature has been enabled or disabled.
   Note: Surge protection works only when both the feature and the service setting are enabled.

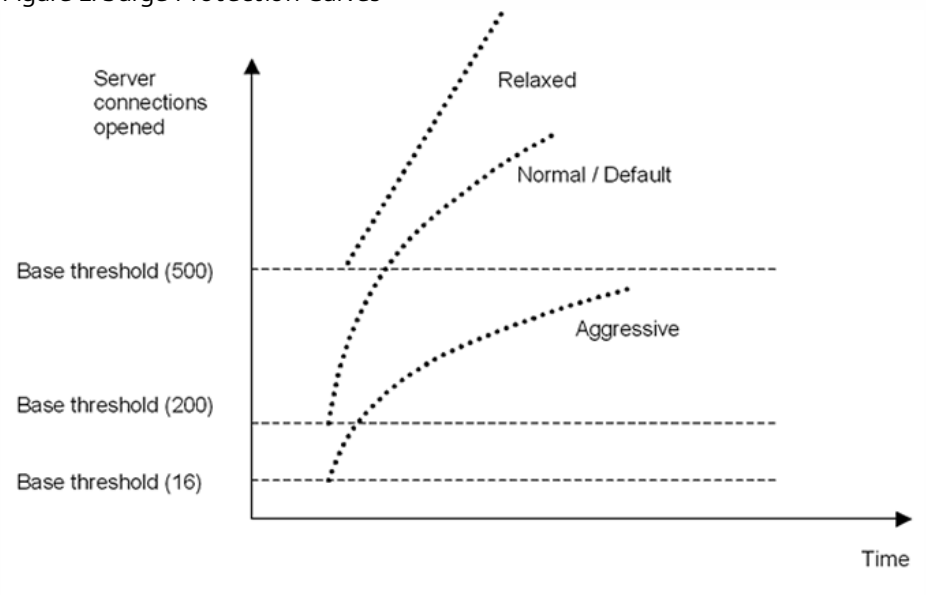
Setting Thresholds for Surge Protection

May 14, 2012

To set the rate at which the NetScaler appliance opens connections to the server, you must configure the threshold and throttle values for surge protection.

The following figure shows the surge protection curves that result from setting the throttle rate to relaxed, normal, or aggressive. Depending on the configuration of the server capacity, you can set base threshold values to generate appropriate surge protection curves.

Figure 1. Surge Protection Curves

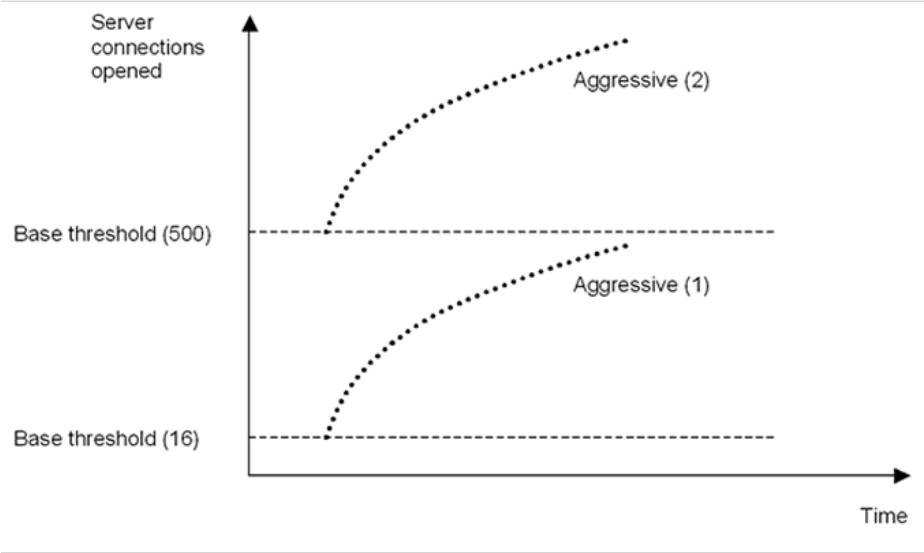


Your configuration settings affect the behavior of surge protection in the following manner:

- If you do not specify a throttle rate, it is set to normal (the default value), and the base threshold is set to 200, as shown in the preceding figure.
- If you specify a throttle rate (aggressive, normal, or relaxed) without specifying a base threshold, the curve reflects the default values of the base threshold for that throttle rate. For example, if you set the throttle rate to relaxed, the resulting curve will have the base threshold value of 500.
- If you specify only the base threshold, the entire surge protection curve shifts up or down, depending on the value you specify, as shown in the figure that follows.
- If you specify both a base threshold and a throttle rate, the resulting surge protection curve is based on the set throttle rate and adjusted according to the value set for the base threshold.

In the following figure, the lower curve (Aggressive 1) results when the throttle rate is set to aggressive but the base threshold is not set. The upper curve (Aggressive 2) results when the base threshold is set to 500, but the throttle rate is not set. The second upper curve (Aggressive 2) also results when the base threshold is set to 500, and the throttle rate is set to aggressive.

Figure 2. Aggressive Rate with the Default or a Set Base Threshold



To set the threshold for surge protection by using the configuration utility

1. In the navigation pane, expand System, and then select Settings.
2. In the details pane, click Global System Settings.
3. If you want to set a base threshold different from the default for the throttle rate, in the Configure Global Settings dialog box, Base Threshold text box, enter the maximum number of concurrent server connections allowed before surge protection is triggered. The base threshold is the maximum number of server connections that can be open before surge protection is activated. The maximum value for this setting is 32,767 server connections. The default setting for this value is controlled by the throttle rate you choose in the next step.
   Note: If you do not set an explicit value here, the default value will be used.
4. In the Throttle drop-down list, select a throttle rate. The throttle is the rate at which the NetScaler appliance allows connections to the server to be opened. The throttle can be set to the following values:
   Aggressive
   Choose this option when the connection-handling and surge-handling capacity of the server is low and the connection needs to be managed carefully. When you set the throttle to aggressive, the base threshold is set to a default value of 16, which means that surge protection is triggered whenever there are 17 or more concurrent connections to the server.
   Normal
   Choose this option when there is no external load balancer behind the NetScaler appliance or downstream. The base threshold is set to a value of 200, which means that surge protection is triggered whenever there are 201 or more concurrent connections to the server. Normal is the default throttle option.
   Relaxed
   Choose this option when the NetScaler appliance is performing load balancing between a large number of Web servers, and can therefore handle a high number of concurrent connections. The base threshold is set to a value of 500, which means that surge protection is triggered only when there are 501 or more concurrent connections to the server.
5. Click OK. A message appears in the status bar, stating that the global settings are configured.

Flushing the Surge Queue

Dec 04, 2013

When a physical server receives a surge of requests, it becomes slow to respond to the clients that are currently connected to it, which leaves users dissatisfied and disgruntled. Often, the overload also causes clients to receive error pages. To avoid such overloads, the NetScaler appliance provides features such as surge protection, which controls the rate at which new connections to a service can be established.

The appliance does connection multiplexing between clients and physical servers. When it receives a client request to access a service on a server, the appliance looks for an already established connection to the server that is free. If it finds a free connection, it uses that connection to establish a virtual link between the client and the server. If it does not find an existing free connection, the appliance establishes a new connection with the server, and establishes a virtual link between client and the server. However, if the appliance cannot establish a new connection with the server, it sends the client request to a surge queue. If all the physical servers bound to the load balancing or content switching virtual server reach the upper limit on client connections (max client value, surge protection threshold or maximum capacity of the service), the appliance cannot establish a connection with any server. The surge protection feature uses the surge queue to regulate the speed at which connections are opened with the physical servers. The appliance maintains a different surge queue for each service bound to the virtual server.

The length of a surge queue increases whenever a request comes for which the appliance cannot establish a connection, and the length decreases whenever a request in the queue gets sent to the server or a request gets timed out and is removed from the queue.

If the surge queue for a service or service group becomes too long, you may want to flush it. You can flush the surge queue of a specific service or service group, or of all the services and service groups bound to a load balancing virtual server. Flushing a surge queue does not affect the existing connections. Only the requests present in the surge queue get deleted. For those requests, the client has to make a fresh request.

You can also flush the surge queue of a content switching virtual server. If a content switching virtual server forwards some requests to a particular load balancing virtual server, and the load balancing virtual server also receives some other requests, when you flush the surge queue of the content switching virtual server, only the requests received from this content switching virtual server are flushed; the other requests in the surge queue of the load balancing virtual server are not flushed.

Note: You cannot flush the surge queues of cache redirection, authentication, VPN or GSLB virtual servers or GSLB services.
Note: Do not use the Surge Protection feature if Use Source IP (USIP) is enabled.

The flush ns surgeQ command works in the following manner:

- You can specify the name of a service, service group, or virtual server whose surge queue has to be flushed.
- If you specify a name while executing the command, surge queue of the specified entity will be flushed. If more than one entity has the same name, the appliance flushes surge queues of all those entities.
- If you specify the name of a service group, and a server name and port while executing the command, the appliance flushes the surge queue of only the specified service group member.
- You cannot directly specify a service group member (<serverName> and <port>) without specifying the name of the service group (<name>) and you cannot specify <port> without a <serverName>. Specify the <serverName> and <port> if you want to flush the surge queue for a specific service group member.
- If you execute the command without specifying any names, the appliance flushes the surge queues of all the entities present on the appliance.
- If a service group member is identified with a server name, you must specify the server name in this command; you cannot specify its IP address.

At the command prompt, type:

flush ns surgeQ [-name <name>] [-serverName <serverName> <port>]

Examples

1.
flush ns surgeQ –name SVC1ANZGB –serverName 10.10.10.1 80
The above command flushes the surge queue of the service or virtual server that is named SVC1ANZGB and has IP address as 10.10.10
2.
flush ns surgeQ
The above command flushes all the surge queues on the appliance.

1. Navigate to Traffic Management > Load Balancing.
2. To select an entity, do one of the following:
   - To flush the surge queue of a virtual server, click Virtual Servers, and then select the virtual server.
   - To flush the surge queue of a service, click Services, and then select the service.

- To flush the surge queue of all the members in a service group, click Service Groups, and then select the service group.
- To flush the surge queue of a specific member in a service group, click Service Groups, and in the action pane, click Manage Members. In the Manage Members of a Service Group dialog box, select the service group member.

Note: You can select multiple entities in any window.

Note: To flush the surge queue of a content switching virtual server, in Steps 1 and 2, expand Content Switching, and then select a virtual server.

3. In the action pane, click Flush Surge Queue.
4. Click OK.

Note: On the appliance, if there are other entities with the same name as you selected, you are alerted that the surge queues of those entities would also be flushed. Take an appropriate action.

API

Mar 14, 2012

The following topics provides information on the API support provided for the NetScaler appliance. Intended for application developers who want to configure and monitor a NetScaler appliance programmatically.

NITRO API	Describes the use of the NITRO APIs for the REST, Java, and .NET platforms.
XML API	Describes the properties and use of the XML API.

NITRO API

Jun 03, 2014

The NetScaler NITRO protocol allows you to configure and monitor the NetScaler appliance programmatically.

NITRO exposes its functionality through Representational State Transfer (REST) interfaces. Therefore, NITRO applications can be developed in any programming language. Additionally, for applications that must be developed in Java or .NET, NITRO APIs are exposed through relevant libraries that are packaged as separate Software Development Kits (SDKs).

Note: You must have a basic understanding of the NetScaler appliance before using NITRO.
To use the NITRO protocol, the client application needs only the following:

- Access to a NetScaler appliance, version 9.2 or later.
- To use REST interfaces, you must have a system to generate HTTP or HTTPS requests (payload in JSON format) to the NetScaler appliance. You can use any programming language or tool.
- For Java clients, you must have a system where Java Development Kit (JDK) 1.5 or later is available. The JDK can be downloaded from http://www.oracle.com/technetwork/java/javase/downloads/index.html.
- For .NET clients, you must have a system with .NET framework 3.5 or later installed. The .NET framework can be downloaded from http://www.microsoft.com/downloads/en/default.aspx.

Note: You can also use XML APIs to configure the NetScaler appliance programmatically. For more information, see "XML API".

Obtaining the NITRO Package

Jun 03, 2014

The NITRO package is available as a tar file on the Downloads page of the NetScaler appliance's configuration utility. You must download and un-tar the file to a folder on your local system. This folder is referred to as <NITRO_SDK_HOME> in this documentation.

The folder contains the NITRO libraries in the lib subfolder. The libraries must be added to the client application classpath to access NITRO functionality. The <NITRO_SDK_HOME> folder also provides samples and documentation that can help you understand the NITRO SDK.

Note:

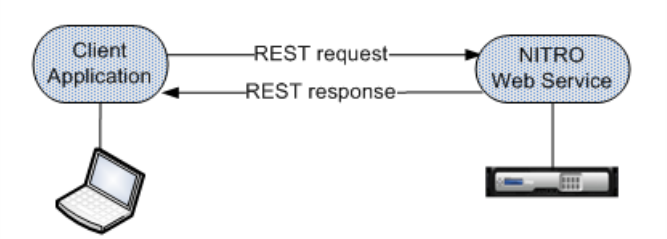
- The REST package contains only documentation for using the REST interfaces.

How NITRO Works

Jun 03, 2014

The NITRO infrastructure consists of a client application and the NITRO Web service running on a NetScaler appliance. The communication between the client application and the NITRO web service is based on REST architecture using HTTP or HTTPS.

Figure 1. NITRO execution flow



As shown in the above figure, a NITRO request is executed as follows:

1. The client application sends REST request message to the NITRO web service. When using the SDKs, an API call is translated into the appropriate REST request message.
2. The web service processes the REST request message.
3. The NITRO web service returns the corresponding REST response message to the client application. When using the SDKs, the REST response message is translated into the appropriate response for the API call.

To minimize traffic on the NetScaler network, you retrieve the whole state of a resource from the server, make modifications to the state of the resource locally, and then upload it back to the server in one network transaction. For example, to update a load balancing virtual server, you must retrieve the object, update the properties, and then upload the changed object in a single transaction.

Note: Local operations on a resource (changing its properties) do not affect its state on the server until the state of the object is explicitly uploaded.
NITRO APIs are synchronous in nature. This means that the client application waits for a response from the NITRO web service before executing another NITRO API.

Java API

Jun 05, 2014

NetScaler NITRO APIs are categorized depending on the scope and purpose of the APIs into system APIs, feature configuration APIs, and feature statistics APIs. Additionally, you can import and export AppExpert applications. You can also troubleshoot NITRO operations.

Note: All NITRO operations are logged in the /var/log/nitro.log file on the appliance.

Tutorials

Nov 05, 2013

These tutorials demonstrate the end-to-end usage of NITRO to achieve the following:

- Create Your First NITRO Application
- Create a NetScaler Cluster

Create Your First NITRO Application

Sep 12, 2012

After completing this tutorial, you will understand and be able to perform the following tasks:

- Integrate NITRO with the IDE
- Log in to the appliance
- Create a load balancing virtual server (lbvserver)
- Retrieve details of an lbvserver
- Delete an lbvserver
- Save the configurations on the appliance
- Log out of the appliance
- Debug the NITRO application

Before you begin, make sure that you have the latest NITRO SDK and that the client application satisfies the prerequisites for using the NITRO SDK.

For the executable code, see the <NITRO_SDK_HOME>/sample/MyFirstNitroApplication.java sample file.

1. Copy the libraries from <NITRO_SDK_HOME>/lib folder to the project classpath.
2. Create a new class and name it **MyFirstNitroApplication**.
3. Create an instance of `com.citrix.netscaler.nitro.service.nitro_service` class. This instance is used to perform all operations on the appliance:
   ```
   nitro_service ns_session = new nitro_service("10.102.29.170","HTTP");
   ```
   This code establishes a connection with an appliance that has IP address 10.102.29.170 and uses the HTTP protocol. Replace 10.102.29.170 with the IP address of the NetScaler appliance that you have access to.
4. Use the `nitro_service` instance to log in to the appliance using your credentials:
   ```
   ns_session.login("admin","verysecret");
   ```
   This code logs into the appliance, with user name as `admin` and password as `verysecret`. Replace the credentials with your login credentials.
5. Enable the load balancing feature:
   ```
   String[] features_to_be_enabled = {"lb"};
   ns_session.enable_features(features_to_be_enabled);
   ```
   This code first sets the features to be enabled in an array and then enables the LB feature.
6. Create an instance of the `com.citrix.netscaler.nitro.resource.config.lb.lbvserver` class. You will use this instance to perform operations on the lbvserver.
   ```
   lbvserver new_lbvserver_obj = new lbvserver();
   ```
7. Use the `lbvserver` instance to create a new lbvserver:
   ```
   new_lbvserver_obj.set_name("MyFirstLbVServer");
   new_lbvserver_obj.set_ipv46("10.102.29.88");
   new_lbvserver_obj.set_servicetype("HTTP");
   new_lbvserver_obj.set_port(88);
   new_lbvserver_obj.set_lbmethod("ROUNDROBIN");
   lbvserver.add(ns_session,new_lbvserver_obj);
   ```
   This code first sets the attributes (name, IP address, service type, port, and load balancing method) of the lbvserver locally and then adds it to the appliance by using the corresponding `add()` method.
8. Retrieve the details of the lbvserver you have created:
   ```
   new_lbvserver_obj = lbvserver.get(ns_session,new_lbvserver_obj.get_name());
   System.out.println("Name : " +new_lbvserver_obj.get_name() +"\n" +"Protocol : " +new_lbvserver_obj.get_servicetype());
   ```
   This code first retrieves the details of the lbvserver as an object from the NetScaler, extracts the required attributes (name and service type) from the object, and displays the results.
9. Delete the lbvserver you created in the above steps:
   ```
   lbvserver.delete(ns_session, new_lbvserver_obj.get_name());
   ```

10. Save the configurations:
    ns_session.save_config();
11. Log out of the appliance:
    ns_session.logout();

All NITRO exceptions are captured by the com.citrix.netscaler.nitro.exception.nitro_exception class. For a more detailed description, see Exception Handling.

Create a NetScaler Cluster

Nov 05, 2013

After completing this tutorial you will be able to create a two-node NetScaler cluster. To add more appliances to the cluster you must repeat the procedure that adds and joins the node to the cluster.

For the executable code, see the <NITRO_SDK_HOME>/sample/CreateCluster.java sample file.

1. Copy the libraries from <NITRO_SDK_HOME>/lib folder to the project classpath.
2. Create a new class and name it CreateCluster.
3. Log on to one of the appliances that you want to add to the cluster and create a cluster:

```
//Connect to the first appliance that you want to add to the cluster
nitro_service nonClipSession0 = new nitro_service(nsipAddress0,protocol);
nonClipSession0.login(uName,password);

//Create a cluster instance
clusterinstance newClusterInstance = new clusterinstance();
newClusterInstance.set_clid(1);
clusterinstance.add(nonClipSession0,newClusterInstance);

//Add the appliance to the cluster
clusternode ClusterNode0 = new clusternode();
ClusterNode0.set_nodeid(0);
ClusterNode0.set_ipaddress(nsipAddress0);
ClusterNode0.set_state("ACTIVE");
ClusterNode0.set_backplane("0/1/1");
clusternode.add(nonClipSession0,ClusterNode0);

//Add the cluster IP address
nsip newNSIPAddress = new nsip();
newNSIPAddress.set_ipaddress(clipAddress);
newNSIPAddress.set_netmask("255.255.255.255");
newNSIPAddress.set_type("CLIP");
nsip.add(nonClipSession0,newNSIPAddress);

//Enable the cluster instance
clusterinstance.enable(nonClipSession0, newClusterInstance);

//Save the configurations
nonClipSession0.save_config();

//Warm reboot the appliance
nonClipSession0.reboot(true);
```

   The cluster is created and the first node is added to the cluster. This node becomes the initial configuration coordinator of the cluster.

4. Log on to the cluster IP address to add other appliances to the cluster:

```
//Connect to the cluster IP address
nitro_service clipSession = new nitro_service(clipAddress,protocol);
clipSession.login(uName,password);

//Add the node to the cluster
clusternode ClusterNode1 = new clusternode();
ClusterNode1.set_nodeid(1);
ClusterNode1.set_ipaddress(nsipAddress1);
ClusterNode1.set_state("ACTIVE");
ClusterNode1.set_backplane("1/1/1");
clusternode.add(clipSession,ClusterNode1);

//Save the configurations
clipSession.save_config();
```

5. Log on to the appliance that you added in the previous step and join it to the cluster:

```
//Connect to the node that you have just added to the cluster
nitro_service nonClipSession1 = new nitro_service(nsipAddress1,protocol);
nonClipSession1.login(uName,password);

//Join the node to the cluster
cluster newCluster = new cluster();
newCluster.set_clip(clipAddress);
newCluster.set_password(password);
cluster.join(nonClipSession1,newCluster);

//Save the configurations
nonClipSession1.save_config();

//Warm reboot the appliance
nonClipSession1.reboot(true);
```

The second node is now a part of the cluster.

6. Verify the details of the cluster by logging on to the cluster IP address

```
//Retrieving the cluster node details
Long id = new Long(1);
clusternode node= clusternode.get(clipSession, id);
System.out.println("Node ID: "+ node.get_nodeid() + " | Admin state: " + node.get_state() + " | Backplane interface: "+ node.get_backplane());

//Retrieving the cluster instance details
Long id1 = new Long(1);
clusterinstance instance= clusterinstance.get(clipSession, id1);
System.out.println("Cluster instance ID: "+ instance.get_clid() + " | Operational state: " +instance.get_operationalstate());
```

System APIs

Jun 03, 2014

The first step towards using NITRO is to establish a session with the NetScaler appliance and then authenticate the session by using the NetScaler administrator's credentials.

You must create an object of the `com.citrix.netscaler.nitro.service.nitro_service` class by specifying the NetScaler IP (NSIP) address and the protocol to connect to the appliance (HTTP or HTTPS). You then use this object and log on to the appliance by specifying the user name and the password of the NetScaler administrator.

Note: You must have a user account on that appliance. The configuration operations that you perform are limited by the administrative roles assigned to your account.
The following sample code establishes a session with a NetScaler appliance with IP address 10.102.29.60 by using the HTTPS protocol:

```
//Specify the NetScaler appliance IP address and protocol
nitro_service ns_session = new nitro_service("10.102.29.60","https");

//Specify the login credentials
ns_session.login("admin","verysecret");
```

Note: When using HTTPS, you must make sure that the root CA is added to the truststore. By default, NITRO validates the SSL certificate and verifies the hostname. To disable this validation, use the following:

```
ns_session.set_certvalidation(false);
ns_session.set_hostnameverification(false);
```

Note: By default, the connection to the appliance expires after 30 minutes of inactivity. You can modify the timeout period by specifying a new timeout period (in seconds) in the `login` method. For example, to modify the timeout period to 60 minutes:

```
ns_session.login("admin","verysecret",3600);
```

You must use the `nitro_service` object in all further NITRO operations on the appliance. For example to save the configurations on the appliance, you must use the `nitro_service` object as follows:

```
ns_session.save_config();
```

The `nitro_service` class also provides APIs to perform other system-level operations such as enabling and disabling NetScaler features and modes, saving and clearing NetScaler configurations, setting the session timeout, setting the severity of the exceptions to be handled, setting the behavior of bulk operations, and disconnecting from the appliance.

Feature Configuration APIs

Jun 03, 2014

NetScaler resources are organized into a set of packages or namespaces. Each package or namespace corresponds to a NetScaler feature. For example, all load-balancing related resources, such as load balancing virtual server, load balancing group, and load balancing monitor are available in com.citrix.netscaler.nitro.resource.config.lb.

Similarly, all application firewall related resources, such as application firewall policy and application firewall archive are available in com.citrix.netscaler.nitro.resource.config.appfw.

Each NetScaler resource is represented by a class. For example, the class that represents a load balancing virtual server is called lbvserver (in com.citrix.netscaler.nitro.resource.config.lb). The state of a resource is represented by properties of a class. You can set the value for these properties by using the set_<propertyname>() methods provided by the resource class. For example to set the IP address of a load balancing virtual server, the lbvserver class provides the set_ipv46() method. Similarly, you can get the value of these properties by using the get_<propertyname>() methods of the resource class.

Note: The setter and getter properties are always executed locally on the client. They do not involve any network interaction with the NITRO web service. All properties have basic simple types: integer, long, boolean, and string.
A resource class provides APIs to perform the following operations:

Create | Retrieve | Update | Delete | Enable/Disable | Unset | Bind/Unbind | Global bind | Bulk operations

Create

To create a new resource, instantiate the resource class, configure the resource by setting its properties locally, and then upload the new resource instance to the NetScaler appliance.

The following sample code creates a load balancing virtual server:

```
//Create an instance of the lbvserver class
lbvserver new_lbvserver_obj = new lbvserver();

//Set the properties of the resource locally
new_lbvserver_obj.set_name("MyFirstLbVServer");
new_lbvserver_obj.set_ipv46("10.102.29.88");
new_lbvserver_obj.set_port(88);
new_lbvserver_obj.set_servicetype("HTTP");
new_lbvserver_obj.set_lbmethod("ROUNDROBIN");

//Upload the resource to NetScaler
lbvserver.add(ns_session,new_lbvserver_obj);
```

Retrieve

To retrieve the properties of a resource, you retrieve the resource object from the NetScaler appliance. Once the object is retrieved, you can extract the required properties of the resource locally, without further network traffic.

The following sample code retrieves the details of a load balancing virtual server:

```
//Retrieve the resource object from the NetScaler
```

```
new_lbvserver_obj = lbvserver.get(ns_session,"MyFirstLbVServer");

//Extract the properties of the resource from the object locally
System.out.println(new_lbvserver_obj.get_name());
System.out.println(new_lbvserver_obj.get_servicetype());
```

You can also retrieve resources by specifying a filter on the value of their properties by using the `com.citrix.netscaler.nitro.util.filtervalue` class.

For example, you can retrieve all the load balancing virtual servers that have their port set to 80 and servicetype to HTTP:

```
filtervalue[] filter = new filtervalue[2];
filter[0] = new filtervalue("port","80");
filter[1] = new filtervalue("servicetype","HTTP");
lbvserver[] result = lbvserver.get_filtered(ns_session,filter);
```

You can also retrieve all NetScaler resources of a certain type, such as all services in the NetScaler appliance, by calling the static `get()` method on the service class, without providing a second parameter, as follows:

```
service[] resources = service.get(ns_session);
```

Update

To update the properties of a resource, instantiate the resource class, specify the name of the resource to be updated, configure the resource by updating its properties locally, and then upload the updated resource instance to the NetScaler appliance.

The following sample code updates the service type and load balancing method of a load balancing virtual server:

```
//Create an instance of the lbvserver class
lbvserver update_lb = new lbvserver();

//Specify the name of the lbvserver to be updated
update_lb.set_name("MyFirstLbVServer");

//Specify the updated service type and lb method
update_lb.set_servicetype("https");
update_lb.set_lbmethod("LEASTRESPONSETIME");

//Upload the resource to NetScaler
lbvserver.update(ns_session,update_lb);
```

Note: Some properties in some NetScaler resources are not allowed to be modified after creation. The port number or the service type (protocol) of a load balancing virtual server or a service, are examples of such properties. Even though the update method appears to succeed, these properties retain their original values on the appliance.

Delete

To delete an existing resource, invoke the `delete()` method on the resource class, by passing the name of the resource.

The following sample code deletes a load balancing virtual server with name "MyFirstLbVServer":

```
lbvserver remove_lb = new lbvserver();
remove_lb.set_name("MyFirstLbVServer");
lbvserver.delete(ns_session, remove_lb);
```

Enable/Disable

To enable a resource, invoke the enable() method.

The following sample code enables a load balancing virtual server named "lb_vip":

```
lbvserver obj = new lbvserver();
obj.set_name = "lb_vip";
lbvserver.enable(ns_session, obj);
```

Note: To disable a resource, invoke the disable() method.

```
lbvserver.disable(ns_session,obj);
```

Unset

To unset the value that is set to a parameter, invoke the unset() method on the resource class, by passing the name of the resource and the parameters to be unset. If the parameter has a default value, the value is reset to that value.

The following sample code unsets the load balancing method and the comments of a load balancing virtual server named "lb_123":

```
lbvserver lb1 = new lbvserver();
lb1.set_name("lb_123");
String args[] = {"comment", "lbmethod"};
lbvserver.unset(ns_session, lb1, args);
```

Bind/Unbind

NetScaler resources form relationships with each other through the process of binding. This is how services are associated with a load balancing virtual server (by binding them to it), or how various policies are bound to a load balancing virtual server. Each binding relationship is represented in NITRO by its own class.

To bind one NetScaler resource to another, you must instantiate the appropriate binding class (for example, to bind a service to a load balancing virtual server, you must instantiate the lbvserver_service_binding class) and add it to the NetScaler configuration (by using the static add() method on this class).

Binding classes have a property representing the name of each resource in the binding relationship. They can also have other properties related to that relationship (for example, the weight of the binding between a load balancing virtual server and a service).

The following sample code binds a service to a load balancing virtual server, by specifying a certain weight for the binding:

```
lbvserver_service_binding bindObj = new lbvserver_service_binding();
bindObj.set_name("MyFirstLbVServer");
bindObj.set_servicename("svc_prod");
bindObj.set_weight(20);
lbvserver_service_binding.add(ns_session,bindObj);
```

Note: To unbind a resource from another, invoke the delete() method from the resource binding class, by passing the name of the two resources.

The following code sample unbinds a service from a server:

```
lbvserver_service_binding bindObj = new lbvserver_service_binding();
bindObj.set_name("MyFirstLbVServer");
bindObj.set_servicename("svc_prod");
lbvserver_service_binding.delete(ns_session,bindObj);
```

Global bind

Some NetScaler resources can be bound globally to affect the whole system. For example, a compression policy can be bound to an load balancing virtual server, in which case the policy affects only the traffic on that load balancing virtual server. However, if bound globally, it can affect any traffic on the appliance, regardless of which virtual servers handle the traffic.

Some NITRO classes can be used to bind resources globally. These classes have names that follow the following pattern: <featurename>global_<resourcetype>_binding.

For example, the class aaaglobal_preauthenticationpolicy_binding is used to bind preauthentication policies globally.

The following sample code creates a preauthentication action and a preauthentication policy that uses that action, and then binds the policy globally at priority 200:

```
aaapreauthenticationaction preauth_act1;
aaapreauthenticationpolicy preauth_pol1;
aaaglobal_aaapreauthenticationpolicy_binding glob_binding;
preauth_act1 = new aaapreauthenticationaction();
preauth_act1.set_name("preauth_act1");
preauth_act1.set_preauthenticationaction("ALLOW");
aaapreauthenticationaction.add(ns_session,preauth_act1);

preauth_pol1 = new aaapreauthenticationpolicy();
preauth_pol1.set_name("preauth_pol1");
preauth_pol1.set_rule("CLIENT.APPLICATION.PROCESS(antivirus.exe) EXISTS");
preauth_pol1.set_reqaction("preauth_act1");
aaapreauthenticationpolicy.add(ns_session,preauth_pol1);

glob_binding = new aaaglobal_aaapreauthenticationpolicy_binding();
glob_binding.set_policy("preauth_pol1");
glob_binding.set_priority(200);
aaaglobal_aaapreauthenticationpolicy_binding.add(ns_session,glob_binding);
```

Bulk operations

You can create, retrieve, update, and delete multiple resources simultaneously and thus minimize network traffic. For example, you can add multiple load balancing virtual servers in the same operation. To perform a bulk operation, you instantiate an array of the resource class, configure the properties of all the instances locally, and then upload all the instances to the NetScaler with one command.

To account for the failure of some operations within the bulk operation, NITRO allows you to configure one of the following behaviors:

- Exit. When the first error is encountered, the execution stops. The commands that were executed before the error are committed.
- Rollback. When the first error is encountered, the execution stops. The commands that were executed before the error are rolled back. Rollback is only supported for add and bind commands.
- Continue. All the commands in the list are executed even if some commands fail.

Note: You must configure the required behavior while establishing a connection with the appliance.

```
nitro_service ns_session = new nitro_service("10.102.29.60","http");
```

```
ns_session.set_onerror(OnerrorEnum.CONTINUE);
ns_session.login("admin","verysecret");
```

The following sample code creates two load balancing virtual servers:

```
//Create an array of lbvserver instances
lbvserver[] lbs = new lbvserver[2];

//Specify properties of the first lbvserver
lbs[0] = new lbvserver();
lbs[0].set_name("lbvserv1");
lbs[0].set_servicetype("http");
lbs[0].set_ipv46("10.70.136.5");
lbs[0].set_port(80);

//Specify properties of the second lbvserver
lbs[1] = new lbvserver();
lbs[1].set_name("lbvserv2");
lbs[1].set_servicetype("https");
lbs[1].set_ipv46("10.70.136.5");
lbs[1].set_port(443);

//Upload the properties of the two lbvservers to the NetScaler
lbvserver.add(ns_session,lbs);
```

Cluster APIs

Jun 03, 2014

For managing clusters, you can add or remove a cluster instance or an individual node and perform a few other instance or node operations such as viewing instance or node properties. You can also configure the cluster IP address. Other cluster-management tasks include joining a NetScaler appliance to the cluster and configuring a linkset.

Cluster Instance Operations

The com.citrix.netscaler.nitro.resource.config.cluster.clusterinstance class provides APIs to manage a cluster instance.

The following sample code creates a cluster instance with ID 1:

```
clusterinstance new_cl_inst_obj = new clusterinstance();
//Set the properties of the cluster instance locally
new_cl_inst_obj.set_clid(1);
new_cl_inst_obj.set_preemption("ENABLED");

//Upload the cluster instance
clusterinstance.add(ns_session,new_cl_inst_obj);
```

Cluster Node Operations

The com.citrix.netscaler.nitro.resource.config.cluster.clusternode class provides APIs to manage cluster nodes.

The following sample code adds a cluster node with NSIP address 10.102.29.60:

```
clusternode new_cl_node_obj = new clusternode();
//Set the properties of the cluster node locally
new_cl_node_obj.set_nodeid(0);
new_cl_node_obj.set_ipaddress("10.102.29.60");
new_cl_node_obj.set_state("ACTIVE");
new_cl_node_obj.set_backplane("0/1/1");

//Upload the cluster node
clusternode.add(ns_session,new_cl_node_obj);
```

Add a Cluster IP Address

The com.citrix.netscaler.nitro.resource.config.ns.nsip class provides the add() API to configure an IP address. To configure the IP address as a cluster IP address, you must specify the type as CLIP.

The following sample code configures a cluster IP address on NetScaler appliance with IP address 10.102.29.60:

```
nsip new_nsip_obj = new nsip();
//Set the properties locally
new_nsip_obj.set_ipaddress("10.102.29.61");
new_nsip_obj.set_netmask("255.255.255.255");
new_nsip_obj.set_type("CLIP");

//Upload the cluster node
```

```
nsip.add(ns_session,new_nsip_obj);
```

Add a Spotted IP Address

The com.citrix.netscaler.nitro.resource.config.ns.nsip class provides the add() API to configure an IP address. To configure the IP address as spotted, you must specify the ID of the node that must own the IP address. This configuration must be done on the cluster IP address.

The following sample code configures a spotted SNIP address on a node with ID 1:

```
nsip new_nsip_obj = new nsip();
//Set the properties locally
new_nsip_obj.set_ipaddress("10.102.29.77");
new_nsip_obj.set_netmask("255.255.255.0");
new_nsip_obj.set_type("SNIP");
new_nsip_obj.set_ownernode(1);

//Upload the cluster node
nsip.add(ns_session,new_nsip_obj);
```

Join NetScaler Appliance to Cluster

The com.citrix.netscaler.nitro.resource.config.cluster.cluster class provides the join() API to join a NetScaler appliance to the cluster. You must specify the cluster IP address and the nsroot password of the configuration coordinator.

The following sample joins a NetScaler appliance to a cluster:

```
cluster new_cl_obj = new cluster();
//Set the properties of the cluster  locally
new_cl_obj.set_clip("10.102.29.61");
new_cl_obj.set_password("verysecret");

//Upload the cluster
cluster.add(ns_session,new_cl_obj);
```

Linkset Operations

The com.citrix.netscaler.nitro.resource.config.network.linkset class provides the APIs to manage linksets.

To configure a linkset, do the following:

1. Add a linkset by invoking the add() method of the linkset class.
2. Bind the interfaces to the linkset using the add() method of the linkset_interface_binding class.

The following sample code creates a linkset LS/1 and bind interfaces 1/1/2 and 2/1/2 to it:

```
//Create the linkset
linkset new_linkset_obj = new linkset();
new_linkset_obj.set_id("LS/1");
linkset.add(ns_session,new_linkset_obj);

//Bind the interfaces to the linkset
linkset_interface_binding new_linkif_obj = new linkset_interface_binding();
new_linkif_obj.set_id("LS/1");
```



```
new_linkif_obj.set_ifnum("1/1/2 2/1/2");
linkset_interface_binding.add(ns_session,new_linkif_obj);
```

Feature Statistics APIs

Jun 03, 2014

The NetScaler appliance collects statistics about the usage of its features and the corresponding resources. You can retrieve these statistics by using NITRO API. The statistics APIs are available in different packages from the configuration APIs.

The APIs to retrieve statistics of NetScaler features are available in packages that have the following pattern: com.citrix.netscaler.nitro.resource.stat.<feature>.

For example, APIs to retrieve statistics of the load balancing virtual server are available in the com.citrix.netscaler.nitro.resource.stat.lb package.

The following sample code retrieves the statistics of a load balancing virtual server and displays some of the statistics returned:

```
lbvserver_stats stats = lbvserver_stats.get(ns_session,"MyFirstLbVServer");
System.out.println(stats.get_curclntconnections());
System.out.println(stats.get_deferredreqrate());
```

Note: Not all NetScaler features and resources have statistic objects associated with them.

AppExpert Application APIs

Jun 03, 2014

To export an AppExpert application, you must instantiate the `com.citrix.netscaler.nitro.resource.config.app.application` class, configure the properties of the AppExpert locally, and then export the AppExpert application.

The following sample code exports an AppExpert application named "MyApp1":

```
application myapp = new application();
myapp.set_appname("MyApp1");
myapp.set_apptemplatefilename("myapp_template");
application.export(ns_session,myapp);
```

You can also import an AppExpert application. You must instantiate the `com.citrix.netscaler.nitro.resource.config.app.application` class, configure the properties of the AppExpert locally, and then import the AppExpert application.

The following sample code imports an AppExpert application named "MyApp1":

```
application myapp = new application();
myapp.set_appname("MyApp1");
myapp.set_apptemplatefilename("myapp_template");
application.Import(ns_session,myapp);
```

Exception Handling

Jun 03, 2014

The status of a NITRO request is captured in the com.citrix.netscaler.nitro.exception.nitro_exception class. This class provides the following details of the exception:

- **Session ID.** The session in which the exception occurred.
- **Severity.** The severity of the exception: error or warning. By default, only errors are captured. To capture warnings, you must set the warning flag to true, while connecting to the appliance.
- **Error code.** The status of the NITRO request. An error code of 0 indicates that the NITRO request is successful. A non-zero error code indicates an error in processing the NITRO request.
- **Error message.** Provides a brief description of the exception.

For a list of error codes, see the errorlisting.html file available in the <NITRO_SDK_HOME>/doc/api_reference folder.

.NET API

Jun 05, 2014

NetScaler NITRO APIs are categorized depending on the scope and purpose of the APIs into system APIs, feature configuration APIs, and feature statistics APIs. Additionally, you can import and export AppExpert applications. You can also troubleshoot NITRO operations.

Note: All NITRO operations are logged in the /var/log/nitro.log file on the appliance.

Tutorials

Nov 05, 2013

These tutorials demonstrate the end-to-end usage of NITRO to achieve the following:

- Create Your First NITRO Application
- Create a NetScaler Cluster

Create Your First NITRO Application

Sep 12, 2012

After completing this tutorial, you will understand and be able to perform the following tasks:

- Integrate NITRO with the IDE
- Log in to the appliance
- Create a load balancing virtual server (lbvserver)
- Retrieve details of an lbvserver
- Delete an lbvserver
- Save the configurations on the appliance
- Log out of the appliance
- Debug the NITRO application

Before you begin, make sure that you have the latest NITRO SDK and that the client application satisfies the prerequisites for using the NITRO SDK.

Sample Code

For the executable code, see the <NITRO_SDK_HOME>/sample/MyFirstNitroApplication.cs sample file.

To create your first NITRO application:

1. Copy the libraries from <NITRO_SDK_HOME>/lib folder to the project classpath.
2. Create a new class and name it **MyFirstNitroApplication**.
3. Create an instance of `com.citrix.netscaler.nitro.service.nitro_service` class. This instance is used to perform all operations on the appliance:
   ```
   nitro_service ns_session = new nitro_service("10.102.29.170", "http");
   ```
   This code establishes a connection with an appliance that has IP address 10.102.29.170 and uses the HTTP protocol. Replace 10.102.29.170 with the IP address of the NetScaler appliance that you have access to.
4. Use the `nitro_service` instance to log in to the appliance using your credentials:
   ```
   ns_session.login("admin","verysecret");
   ```
   This code logs into the appliance, with user name as `admin` and password as `verysecret`. Replace the credentials with your login credentials.
5. Enable the load balancing feature:
   ```
   String[] features_to_be_enabled = {"lb"};
   ns_session.enable_features(features_to_be_enabled);
   ```
   This code enables load balancing on the appliance.
6. Create an instance of the `com.citrix.netscaler.nitro.resource.config.lb.lbvserver` class. You will use this instance to perform operations on the lbvserver.
   ```
   lbvserver new_lbvserver_obj = new lbvserver();
   ```
7. Use the `lbvserver` instance to create a new lbvserver:
   ```
   new_lbvserver_obj.name = "MyFirstLbVServer";
   new_lbvserver_obj.ipv46 = "10.102.29.88";
   new_lbvserver_obj.servicetype = "HTTP";
   new_lbvserver_obj.port = 80;
   new_lbvserver_obj.lbmethod = "ROUNDROBIN";
   lbvserver.add(ns_session,new_lbvserver_obj);
   ```
   This code first sets the attributes (name, IP address, service type, port, and load balancing method) of the lbvserver locally and then adds it to the appliance by using the corresponding `add()` method.
8. Retrieve the details of the lbvserver you have created:
   ```
   lbvserver new_lbvserver_obj1 = lbvserver.get(ns_session,new_lbvserver_obj.name);
   System.Console.Out.WriteLine("Name : " +new_lbvserver_obj1.name +"\n" +"Protocol : " +new_lbvserver_obj1.servicetype);
   ```
   This code first retrieves the details of the lbvserver as an object from the NetScaler, extracts the required attributes (name and service type) from the object, and displays the results.
9. Delete the lbvserver you created in the above steps:
   ```
   lbvserver.delete(ns_session, new_lbvserver_obj.name);
   ```
10. Save the configurations:

```
ns_session.save_config();
```

11. Log out of the appliance:

```
ns_session.logout();
```

Debug the NITRO application

All NITRO exceptions are captured by the `com.citrix.netscaler.nitro.exception.nitro_exception` class. For a more detailed description, see Exception Handling.

Create a NetScaler Cluster

Nov 05, 2013

After completing this tutorial you will be able to create a two-node NetScaler cluster. To add more appliances to the cluster you must repeat the procedure that adds and joins the node to the cluster.

Sample Code

For the executable code, see the <NITRO_SDK_HOME>/sample/CreateCluster.cs sample file.

To create a cluster

1. Copy the libraries from <NITRO_SDK_HOME>/lib folder to the project classpath.
2. Create a new class and name it CreateCluster.
3. Log on to one of the appliances that you want to add to the cluster and create a cluster:

```
//Connect to the first appliance that you want to add to the cluster
nitro_service nonClipSession0 = new nitro_service(nsipAddress0,protocol);
nonClipSession0.login(uName,password);

//Create a cluster instance
clusterinstance newClusterInstance = new clusterinstance();
newClusterInstance.clid = 1;
clusterinstance.add(nonClipSession0,newClusterInstance);

//Add the appliance to the cluster
clusternode ClusterNode0 = new clusternode();
ClusterNode0.nodeid = 0;
ClusterNode0.ipaddress = nsipAddress0;
ClusterNode0.state = "ACTIVE";
ClusterNode0.backplane = "0/1/1";
clusternode.add(nonClipSession0,ClusterNode0);

//Add the cluster IP address
nsip newNSIPAddress = new nsip();
newNSIPAddress.ipaddress = clipAddress;
newNSIPAddress.netmask = "255.255.255.255";
newNSIPAddress.type = "CLIP";
nsip.add(nonClipSession0,newNSIPAddress);

//Enable the cluster instance
clusterinstance.enable(nonClipSession0, newClusterInstance);

//Save the configurations
nonClipSession0.save_config();

//Warm reboot the appliance
nonClipSession0.reboot(true);
```

   The cluster is created and the first node is added to the cluster. This node becomes the initial configuration coordinator of the cluster.

4. Log on to the cluster IP address to add other appliances to the cluster:

```
//Connect to the cluster IP address
nitro_service clipSession = new nitro_service(clipAddress,protocol);
clipSession.login(uName,password);

//Add the node to the cluster
clusternode ClusterNode1 = new clusternode();
ClusterNode1.nodeid = 1;
ClusterNode1.ipaddress = nsipAddress1;
ClusterNode1.state = "ACTIVE";
ClusterNode1.backplane = "1/1/1";
clusternode.add(clipSession,ClusterNode1);

//Save the configurations
```

```
clipSession.save_config();
```

5. Log on to the appliance that you added in the previous step and join it to the cluster:

```
//Connect to the node that you have just added to the cluster
nitro_service nonClipSession1 = new nitro_service(nsipAddress1,protocol);
nonClipSession1.login(uName,password);

//Join the node to the cluster
cluster newCluster = new cluster();
newCluster.clip = clipAddress;
newCluster.password = password;
cluster.join(nonClipSession1,newCluster);

//Save the configurations
nonClipSession1.save_config();

//Warm reboot the appliance
nonClipSession1.reboot(true);
```

The second node is now a part of the cluster.

6. Verify the details of the cluster by logging on to the cluster IP address

```
//Retrieving the cluster node details
uint id = 1;
clusternode node= clusternode.get(clipSession, id);
System.Console.Out.WriteLine("Node ID: " + node.nodeid + " | Admin state: " + node.state + " | Backplane interface: " + node.backplane);

//Retrieving the cluster instance details
uint id1 = 1;
clusterinstance instance= clusterinstance.get(clipSession, id1);
System.Console.Out.WriteLine("Cluster instance ID: "+ instance.clid + " | Operational state: " +instance.operationalstate);
```

System APIs

Jun 03, 2014

The first step towards using NITRO is to establish a session with the NetScaler appliance and then authenticate the session by using the NetScaler administrator's credentials.

You must create an object of the com.citrix.netscaler.nitro.service.nitro_service class by specifying the NetScaler IP (NSIP) address and the protocol to connect to the appliance (HTTP or HTTPS). You then use this object and log on to the appliance by specifying the user name and the password of the NetScaler administrator.

Note: You must have a user account on that appliance. The configuration operations that you perform are limited by the administrative roles assigned to your account.
The following sample code establishes a session with a NetScaler appliance with IP address 10.102.29.60 by using the HTTPS protocol:

```
//Specify the NetScaler appliance IP address and protocol
nitro_service ns_session = new nitro_service("10.102.29.60","https");

//Specify the login credentials
ns_session.login("admin","verysecret");
```

Note: By default, the connection to the appliance expires after 30 minutes of inactivity. You can modify the timeout period by specifying a new timeout period (in seconds) in the login method. For example, to modify the timeout period to 60 minutes:

```
ns_session.login("admin","verysecret",3600);
```

You must use the nitro_service object in all further NITRO operations on the appliance. For example to save the configurations on the appliance, you must use the nitro_service object as follows:

```
ns_session.save_config();
```

The nitro_service class also provides APIs to perform other system-level operations such as enabling and disabling NetScaler features and modes, saving and clearing NetScaler configurations, setting the session timeout, setting the severity of the exceptions to be handled, setting the behavior of bulk operations, and disconnecting from the appliance.

Feature Configuration APIs

Jun 03, 2014

NetScaler resources are organized into a set of packages or namespaces. Each package or namespace corresponds to a NetScaler feature. For example, all load-balancing related resources, such as load balancing virtual server, load balancing group, and load balancing monitor are available in com.citrix.netscaler.nitro.resource.config.lb.

Similarly, all application firewall related resources, such as application firewall policy and application firewall archive are available in com.citrix.netscaler.nitro.resource.config.appfw.

Each NetScaler resource is represented by a class. For example, the class that represents a load balancing virtual server is called lbvserver (in com.citrix.netscaler.nitro.resource.config.lb). The state of a resource is represented by properties of a class. You can get and set the properties of the class.

Note: The setter and getter properties are always executed locally on the client. They do not involve any network interaction with the NITRO web service. All properties have basic simple types: integer, long, boolean, and string.
A resource class provides APIs to perform the following operations:

Create | Retrieve | Update | Delete | Enable/Disable | Unset | Bind/Unbind | Global bind | Bulk operations

Create

To create a new resource, instantiate the resource class, configure the resource by setting its properties locally, and then upload the new resource instance to the NetScaler appliance.

The following sample code creates a load balancing virtual server:

```
//Create an instance of the lbvserver class
lbvserver new_lbvserver_obj = new lbvserver();

//Set the properties of the resource locally
new_lbvserver_obj.name = "MyFirstLbVServer";
new_lbvserver_obj.ipv46 = "10.102.29.88";
new_lbvserver_obj.port = 88;
new_lbvserver_obj.servicetype = "HTTP";
new_lbvserver_obj.lbmethod = "ROUNDROBIN";

//Upload the resource to NetScaler
lbvserver.add(ns_session,new_lbvserver_obj);
```

Retrieve

To retrieve the properties of a resource, retrieve the resource object from the NetScaler appliance. Once the object is retrieved, you can extract the required properties of the resource locally, without incurring further network traffic.

The following sample code retrieves the details of a load balancing virtual server:

```
//Retrieve the resource object from the NetScaler
new_lbvserver_obj = lbvserver.get(ns_session,"MyFirstLbVServer");

//Extract the properties of the resource from the object locally
```

```
Console.WriteLine(new_lbvserver_obj.name);
Console.WriteLine(new_lbvserver_obj.servicetype);
```

You can also retrieve resources by specifying a filter on the value of their properties by using the `com.citrix.netscaler.nitro.util.filtervalue` class.

For example, you can retrieve all the load balancing virtual servers that have their port set to 80 and servicetype to HTTP:

```
filtervalue[] filter = new filtervalue[2];
filter[0] = new filtervalue("port","80");
filter[1] = new filtervalue("servicetype","HTTP");
lbvserver[] result = lbvserver.get_filtered(ns_session,filter);
```

You can also retrieve all NetScaler resources of a certain type, such as all services in the NetScaler appliance, by calling the static `get()` method on the service class, without providing a second parameter, as follows:

```
service[] resources = service.get(ns_session);
```

Update

To update the properties of a resource, instantiate the resource class, specify the name of the resource to be updated, configure the resource by updating its properties locally, and then upload the updated resource instance to the NetScaler appliance.

The following sample code updates the service type and load balancing method of a load balancing virtual server:

```
//Create an instance of the lbvserver class
lbvserver update_lb = new lbvserver();

//Specify the name of the lbvserver to be updated
update_lb.name = "MyFirstLbVServer";

//Specify the updated service type and lb method
update_lb.servicetype = "https";
update_lb.lbmethod = "LEASTRESPONSETIME";

//Upload the resource to NetScaler
lbvserver.update(ns_session, update_lb);
```

Note: Some properties in some NetScaler resources are not allowed to be modified after creation. The port number or the service type (protocol) of a load balancing virtual server or a service, are examples of such properties. Even though the update method appears to succeed, these properties retain their original values on the appliance.

Delete

To delete an existing resource, invoke the static method delete() on the resource class, by passing the name of the resource.

The following sample code deletes a load balancing virtual server with name "MyFirstLbVServer":

```
lbvserver remove_lb = new lbvserver();
remove_lb.name("MyFirstLbVServer");
lbvserver.delete(ns_session, remove_lb);
```

Enable/Disable

To enable a resource, invoke the `enable()` method.

The following sample code enables a load balancing virtual server named "lb_vip":

```
lbvserver obj = new lbvserver();
obj.name = "lb_vip";
lbvserver.enable(ns_session, obj);
```

Note: To disable a resource, invoke the disable() method.

```
lbvserver.disable(ns_session, obj);
```

Unset

To unset the value that is set to a parameter, invoke the unset() method on the resource class, by passing the name of the resource and the parameters to be unset. If the parameter has a default value, the value is reset to that value.

The following sample code unsets the load balancing method and the comments of a load balancing virtual server named "lb_123":

```
lbvserver obj = new lbvserver();
obj.name = "lb_123";
String[] args = { "lbmethod","comment" };
lbvserver.unset(ns_session, lb1, args);
```

Bind/Unbind

NetScaler resources form relationships with each other through the process of binding. This is how services are associated with a load balancing virtual server (by binding them to it), or how various policies are bound to a load balancing virtual server. Each binding relationship is represented in NITRO by its own class.

To bind one NetScaler resource to another, you must instantiate the appropriate binding class (for example, to bind a service to a load balancing virtual server, you must instantiate the lbvserver_service_binding class) and add it to the NetScaler configuration (by using the static add() method on this class).

Binding classes have a property representing the name of each resource in the binding relationship. They can also have other properties related to that relationship (for example, the weight of the binding between a load balancing virtual server and a service).

The following sample code binds a service to a load balancing virtual server, by specifying a certain weight for the binding:

```
lbvserver_service_binding bindObj = new lbvserver_service_binding();
bindObj.name = "MyFirstLbVServer";
bindObj.servicename = "svc_prod";
bindObj.weight = 20;
lbvserver_service_binding.add(ns_session,bindObj);
```

Note: To unbind a resource from another, invoke the delete() method from the resource binding class, by passing the name of the two resources.

The following code sample unbinds a service from a server:

```
lbvserver_service_binding bindObj = new lbvserver_service_binding();
bindObj.name("MyFirstLbVServer");
bindObj.servicename("svc_prod");
lbvserver_service_binding.delete(ns_session,bindObj);
```

Global bind

Some NetScaler resources can be bound globally to affect the whole system. For example, a compression policy can be

bound to an load balancing virtual server, in which case the policy affects only the traffic on that load balancing virtual server. However, if bound globally, it can affect any traffic on the appliance, regardless of which virtual servers handle the traffic.

Some NITRO classes can be used to bind resources globally. These classes have names that follow the following pattern: <featurename>global_<resourcetype>_binding.

For example, the class aaaglobal_preauthenticationpolicy_binding is used to bind preauthentication policies globally.

The following sample code creates a preauthentication action and a preauthentication policy that uses that action, and then binds the policy globally at priority 200:

```
aaapreauthenticationaction preauth_act1;
aaapreauthenticationpolicy preauth_pol1;
aaaglobal_aaapreauthenticationpolicy_binding glob_binding;
preauth_act1 = new aaapreauthenticationaction();
preauth_act1.name = "preauth_act1";
preauth_act1.preauthenticationaction = "ALLOW";
aaapreauthenticationaction.add(ns_session, preauth_act1);

preauth_pol1 = new aaapreauthenticationpolicy();
preauth_pol1.name = "preauth_pol1";
preauth_pol1.rule = "CLIENT.APPLICATION.PROCESS(antivirus.exe) EXISTS";
preauth_pol1.reqaction = "preauth_act1";
aaapreauthenticationpolicy.add(ns_session, preauth_pol1);

glob_binding = new aaaglobal_aaapreauthenticationpolicy_binding();
glob_binding.policy = "preauth_pol1";
glob_binding.priority = 200;
aaaglobal_aaapreauthenticationpolicy_binding.add(ns_session,glob_binding);
```

Bulk operations

You can create, retrieve, update, and delete multiple resources simultaneously and thus minimize network traffic. For example, you can add multiple load balancing virtual servers in the same operation. To perform a bulk operation, you instantiate an array of the resource class, configure the properties of all the instances locally, and then upload all the instances to the NetScaler with one command.

To account for the failure of some operations within the bulk operation, NITRO allows you to configure one of the following behaviors:

- **Exit.** When the first error is encountered, the execution stops. The commands that were executed before the error are committed.
- **Rollback.** When the first error is encountered, the execution stops. The commands that were executed before the error are rolled back. Rollback is only supported for add and bind commands.
- **Continue.** All the commands in the list are executed even if some commands fail.

Note: You must configure the required behavior while establishing a connection with the appliance.

```
nitro_service ns_session = new nitro_service("10.102.29.60","http");
ns_session.onerror = OnerrorEnum.CONTINUE;
ns_session.login("admin","verysecret");
```

The following sample code creates two load balancing virtual servers:

```
//Create an array of lbvserver instances
lbvserver[] lbs = new lbvserver[2];

//Specify details of first lbvserver
lbs[0] = new lbvserver();
lbs[0].name = "lbvserv1";
lbs[0].servicetype = "http";
lbs[0].ipv46 = "10.70.136.5";
lbs[0].port = 80;

//Specify details of second lbvserver
lbs[1] = new lbvserver();
lbs[1].name = "lbvserv2";
lbs[1].servicetype = "https";
lbs[1].ipv46 = "10.70.136.5";
lbs[1].port = 443;

//upload the details of the lbvservers to the NITRO server
lbvserver.add(ns_session,lbs);
```

Cluster APIs

Nov 05, 2013

For managing clusters, you can add or remove a cluster instance or an individual node and perform a few other instance or node operations such as viewing instance or node properties. You can also configure the cluster IP address. Other cluster-management tasks include joining a NetScaler appliance to the cluster and configuring a linkset.

Cluster Instance Operations

The com.citrix.netscaler.nitro.resource.config.cluster.clusterinstance class provides APIs to manage a cluster instance.

The following sample code creates a cluster instance with ID 1:

```
clusterinstance new_cl_inst_obj = new clusterinstance();
//Set the properties of the cluster instance locally
new_cl_inst_obj.clid = 1;
new_cl_inst_obj.preemption = "ENABLED";

//Upload the cluster instance
clusterinstance.add(ns_session,new_cl_inst_obj);
```

Cluster Node Operations

The com.citrix.netscaler.nitro.resource.config.cluster.clusternode class provides APIs to manage cluster nodes.

The following sample code adds a cluster node with NSIP address 10.102.29.60:

```
clusternode new_cl_node_obj = new clusternode();
//Set the properties of the cluster node locally
new_cl_node_obj.nodeid = 0;
new_cl_node_obj.ipaddress = "10.102.29.60";
new_cl_node_obj.state = "ACTIVE";
new_cl_node_obj.backplane = "0/1/1";

//Upload the cluster node
clusternode.add(ns_session,new_cl_node_obj);
```

Add a Cluster IP Address

The com.citrix.netscaler.nitro.resource.config.ns.nsip class provides the add() API to configure an IP address. To configure the IP address as a cluster IP address, you must specify the type as CLIP.

The following sample code configures a cluster IP address on NetScaler appliance with IP address 10.102.29.60:

```
nsip new_nsip_obj = new nsip();
//Set the properties locally
new_nsip_obj.ipaddress = "10.102.29.61";
new_nsip_obj.netmask = "255.255.255.255";
new_nsip_obj.type = "CLIP";

//Upload the cluster node
```

```
nsip.add(ns_session,new_nsip_obj);
```

Add a Spotted IP Address

The com.citrix.netscaler.nitro.resource.config.ns.nsip class provides the add() API to configure an IP address. To configure the IP address as spotted, you must specify the ID of the node that must own the IP address. This configuration must be done on the cluster IP address.

The following sample code configures a spotted SNIP address on a node with ID 1:

```
nsip new_nsip_obj = new nsip();
//Set the properties locally
new_nsip_obj.ipaddress = "10.102.29.77";
new_nsip_obj.netmask = "255.255.255.0";
new_nsip_obj.type = "SNIP";
new_nsip_obj.ownernode = 1;

//Upload the cluster node
nsip.add(ns_session,new_nsip_obj);
```

Join NetScaler Appliance to Cluster

The com.citrix.netscaler.nitro.resource.config.cluster.cluster class provides the join() API to join a NetScaler appliance to the cluster. You must specify the cluster IP address and the nsroot password of the configuration coordinator.

The following sample code joins a NetScaler appliance to a cluster:

```
cluster new_cl_obj = new cluster();
//Set the properties of the cluster locally
new_cl_obj.clip = "10.102.29.61";
new_cl_obj.password = "verysecret";

//Upload the cluster node
cluster.add(ns_session,new_cl_node_obj);
```

Linkset Operations

The com.citrix.netscaler.nitro.resource.config.network.linkset class provides the APIs to manage linksets.

To configure a linkset, do the following:

1. Add a linkset by invoking the add() method of the linkset class.
2. Bind the interfaces to the linkset using the add() method of the linkset_interface_binding class.

The following sample code creates a linkset LS/1 and bind interfaces 1/1/2 and 2/1/2 to it:

```
//Create the linkset
linkset new_linkset_obj = new linkset();
new_linkset_obj.id = "LS/1";
linkset.add(ns_session,new_linkset_obj);

//Bind the interfaces to the linkset
linkset_interface_binding new_linkif_obj = new linkset_interface_binding();
new_linkif_obj.id = "LS/1";
```

```
new_linkif_obj.ifnum = "1/1/2 2/1/2";
linkset_interface_binding.add(ns_session,new_linkif_obj);
```

Feature Statistics APIs

Mar 14, 2012

The NetScaler appliance collects statistics about the usage of its features and the corresponding resources. You can retrieve these statistics by using NITRO API. The statistics APIs are available in different namespaces from the configuration APIs.

The APIs to retrieve statistics of NetScaler features are available in namespaces that have the following pattern: com.citrix.netscaler.nitro.resource.stat.<feature>.

For example, APIs to retrieve statistics of the load balancing virtual server are available in the com.citrix.netscaler.nitro.resource.stat.lb namespace.

The following sample code retrieves the statistics of a load balancing virtual server and displays some of the statistics returned:

```
lbvserver_stats stats = lbvserver_stats.get(ns_session,"MyFirstLbVServer");
Console.WriteLine(stats.curclntconnections);
Console.WriteLine(stats.deferredreqrate);
```

Note: Not all NetScaler features and resources have statistic objects associated with them.

AppExpert Application APIs

Mar 14, 2012

To export an AppExpert application, you must instantiate the `com.citrix.netscaler.nitro.resource.config.app.application` class, configure the properties of the AppExpert locally, and then export the AppExpert application.

The following sample code exports an AppExpert application named "MyApp1":

```
application myapp = new application();
myapp.appname = "MyApp1";
myapp.apptemplatefilename = "myapp_template";
application.export(ns_session,myapp);
```

You can also import an AppExpert application. You must instantiate the `com.citrix.netscaler.nitro.resource.config.app.application` class, configure the properties of the AppExpert locally, and then import the AppExpert application.

The following sample code imports an AppExpert application named "MyApp1":

```
application myapp = new application();
myapp.appname = "MyApp1";
myapp.apptemplatefilename = "myapp_template";
application.Import(ns_session,myapp);
```

Exception Handling

Mar 14, 2012

The status of a NITRO request is captured in the com.citrix.netscaler.nitro.exception.nitro_exception class. This class provides the following details of the exception:

- **Session ID.** The session in which the exception occurred.
- **Severity.** The severity of the exception: error or warning. By default, only errors are captured. To capture warnings, you must set the warning flag to true, while connecting to the appliance.
- **Error code.** The status of the NITRO request. An error code of 0 indicates that the NITRO request is successful. A non-zero error code indicates an error in processing the NITRO request.
- **Error message.** Provides a brief description of the exception.

For a list of error codes, see the errorlisting.html file available in the <NITRO_SDK_HOME>/doc/api_reference folder.

REST Web Services

Jun 05, 2014

REST (REpresentational State Transfer) is an architectural style based on simple HTTP requests and responses between the client and the server. REST is used to query or change the state of objects on the server side. In REST, the server side is modeled as a set of entities where each entity is identified by a unique URL. For example, the load balancing virtual server entity is identified by the URL http://<NSIP>/nitro/v1/config/<lbvserver>/<lbvserver_name>.

Each resource also has a state on which the following operations can be performed:

- **Create.** Clients can create new server-side resources on a "container" resource. You can think of container resources as folders, and child resources as files or subfolders. The calling client provides the state for the resource to be created. The state can be specified in the request by using XML or JSON format. The client can also specify the unique URL that will identify the new object. Alternatively, the server can choose and return a unique URL identifying the created object. The HTTP method used for Create requests is POST.
- **Read.** Clients can retrieve the state of a resource by specifying its URL with the HTTP GET method. The response message contains the resource state, expressed in JSON format.
- **Update.** You can update the state of an existing resource by specifying the URL that identifies that object and its new state in JSON or XML, using the PUT HTTP method.
- **Delete.** You can destroy a resource that exists on the server-side by using the DELETE HTTP method and the URL identifying the resource to be removed.

In addition to these four CRUD operations (Create, Read, Update, and Delete), resources can support other operations or actions. These operations use the HTTP POST method, with the URL specifying the operation to be performed and the request body specifying the parameters for that operation.

NetScaler NITRO APIs are categorized depending on the scope and purpose of the APIs into system APIs, feature configuration APIs, and feature statistics APIs.

Note: All NITRO operations are logged in the /var/log/nitro.log file on the appliance.

Performing System Level Operations

Jun 03, 2014

The first step towards using NITRO is to establish a session with the NetScaler appliance and then authenticate the session by using the NetScaler administrator's credentials. You must specify the username and password in the login object. The session ID that is created must be specified in the request header of all further operations in the session.

Note: You must have a user account on the appliance to log on to it. The configuration operations that you can perform are limited by the administrative roles assigned to your account.

To connect to a NetScaler appliance with NSIP address 10.102.29.60 by using the HTTP protocol:

- **URL.** https://10.102.29.60/nitro/v1/config/login/
- **Method.** POST
- **Request.**
  - **Header.**
    Content-Type:application/vnd.com.citrix.netscaler.login+json
    Note: Content types such as 'application/x-www-form-urlencoded' that were supported in earlier versions of NITRO can also be used. You must make sure that the payload is the same as used in earlier versions. The payloads provided in this documentation are only applicable if the content type is of the form 'application/vnd.com.citrix.netscaler.login+json'.
  - **Payload.**

```
{
   "login":
   {
      "username":"admin",
      "password":"verysecret"
   }
}
```

- **Response.**
  - **Header.**
    HTTP/1.0 201 Created
    Set-Cookie:
    NITRO_AUTH_TOKEN=##87305E9C51B06C848F0942; path=/nitro/v1

Note: By default, the connection to the appliance expires after 30 minutes of inactivity. You can modify the timeout period by specifying a new timeout period (in seconds) in the login object. For example, to modify the timeout period to 60 minutes, the request payload is:

```
{
   "login":
   {
      "username":"admin",
      "password":"verysecret",
      "timeout":3600
   }
}
```

You can also connect to the appliance to perform a single operation, by specifying the username and password in the request header of the operation. For example, to connect to an appliance while adding a load balancing virtual server:

- **URL.** https://10.102.29.60/nitro/v1/config/lbvserver/

- **Method.** POST
- **Request.**
  - **Header.**
    X-NITRO-USER:admin
    X-NITRO-PASS:verysecret
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver
    +json
  - **Payload.**
    ```
    {
     "lbvserver":
     {
       ...
       ...
       ...
    }
    ```
- **Response.**
  - **Header.**
    HTTP/1.0 201 Created

You can also perform other system-level operations such as enabling NetScaler features and modes, saving and clearing NetScaler configurations, setting the session timeout, setting the severity of the exceptions to be handled, setting the behavior of bulk operations, and disconnecting from the appliance.

For more information on the REST messages, see the Configuration node of the <NITRO_SDK_HOME>/index.html file.

Example 1: Enable the load balancing feature

- **URL.** http://10.102.29.60/nitro/v1/config/nsfeature?action=enable
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.nsfeature+json
  - **Payload**
    ```
    {
      "nsfeature":
      {
        "feature":
        [
          "LB",
        ]
      }
    }
    ```

Example 2: Save NetScaler configurations

- **URL.** http://10.102.29.60/nitro/v1/config/nsconfig?action=save
- **HTTP Method.** POST
- **Request.**
  - **Header**

Cookie:NITRO_AUTH_TOKEN=tokenvalue
Content-Type:application/vnd.com.citrix.netscaler.nsconfig+json

- **Payload**

```
{
   "nsconfig":{}
}
```

Example 3: Disconnecting from the appliance

- **URL.** https://10.102.29.60/nitro/v1/config/logout/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.logout+json
  - **Payload**

```
{
   "logout":{}
}
```

Note: Make sure that you have saved the configurations before performing this operation.

Configuring NetScaler Features

Jun 05, 2014

A NetScaler appliance has multiple features, and each feature has multiple resources. Each NetScaler resource, depending on the operation to be performed on it, has a unique URL associated with it. URLs for configuration operations have the format http://<NSIP>/nitro/v1/config/<resource_type>/<resource_name>. For example, to access the lbvserver named MyFirstLbVServer on a NetScaler with IP 10.102.29.60, the URL is http://10.102.29.60/nitro/v1/config/lbvserver/MyFirstLbVServer.

Using NITRO you can perform the following operations:

Create | Retrieve | Update | Delete | Enable/Disable | Unset | Bind/Unbind | Bulk operations

For more information on the REST messages, see the Configuration node of the <NITRO_SDK_HOME>/index.html file.

Create

To create a new resource (for example, an lbvserver) on the appliance, specify the resource name and other related arguments in the specific resource object. For a lbvserver resource, the object would be an lbvserver object.

To create an lbvserver named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver+json
  - **Payload**

```
{
  "lbvserver":
  {
    "name":"MyFirstLbVServer",
    "servicetype":"http"
  }
}
```

Retrieve

NetScaler resource properties can be retrieved as follows:

- To retrieve details of all resources of a specific type, specify the resource type in the URL.
  URL format: http://<NSIP>/nitro/v1/config/<resource_type>
- To retrieve details of a specific resource on the NetScaler appliance, specify the resource name in the URL.
  URL format: http://<NSIP>/nitro/v1/config/<resource_type>/<resource_name>
- To retrieve specific details of a resource, specify the resource details that you want to view in the URL.
  URL format: http://<NSIP>/nitro/v1/config/<resource_type>/<resource_name>?attrs=<attrib1>, <attrib2>

- To retrieve details of resources on the basis of some filter, specify the filter conditions in the URL.
  URL format: http://<NSIP>/nitro/v1/config/<resource_type>?filter=<attrib1>:<value>,<attrib2>:<value>

- If the request is likely to result in a large number of resources, you can divide the results into pages and retrieve them page by page.
  For example, assume that you have a NetScaler that has 53 lbvservers and you want to retrieve all the lbvservers. So, instead of retrieving all 53 in one response, you can configure the results to be divided into pages of 10 lbvservers each (6 pages total), and retrieve them from the NetScaler page by page.

  URL format: http://<NSIP>/nitro/v1/config/<resource_type>?pageno=<value>&pagesize=<value>

  You specify the page count with the pagesize parameter and the page number that you want to retrieve with the pageno parameter.

- To get the number of resources that are likely to be returned by a request, you can use the count query string parameter to ask for a count of the resources to be returned, rather than the resources themselves.
  URL format: http://<NSIP>/nitro/v1/config/<resource_type>?count=yes

To retrieve the details of an lbvserver named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver/MyFirstLbVServer/
- **HTTP Method.** GET
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
- **Response.**
  - **Header**
    HTTP/1.0 200 OK
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver+json
  - **Payload**

```
{
  "lbvserver":
  [
    {
      "name":"MyFirstLbVServer",
      "servicetype":"http",
      "insertvserveripport":"OFF",
      "ip":"0.0.0.0",
      "port":80,
      ...
    }
  ]
}
```

Update

To update the details of an existing resource on the NetScaler appliance, specify the resource name, and the arguments to be updated, in the specific resource object.

To change the load balancing method to ROUNDROBIN and update the comment property for a load balancing virtual server named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver/MyFirstLbVServer/
- **HTTP Method.** PUT
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver+json
  - **Payload**

```
{
   "lbvserver":
   {
      "name":"MyFirstLbVServer",
      "lbmethod":"ROUNDROBIN",
      "comment":"Updated comments"
   }
}
```

Delete

To delete a NetScaler resource, specify the resource name in the URL.

To delete a load balancing virtual server named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver/MyFirstLbVServer
- **HTTP Method.** DELETE

Enable/Disable

To enable a resource on the NetScaler appliance, specify the resource name in the specific resource object.

To enable a load balancing virtual server named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver?action=enable
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver+json
  - **Payload**

```
{
   "lbvserver":
   {
      "name":"MyFirstLbVServer"
   }
}
```

Note: To disable a resource, in the URL specify the action as "disable".

Unset

To unset the value that is set to a parameter, specify the action as "unset" and in the payload, specify the parameters to

be unset.

To unset the load balancing method and the comments specified for a load balancing virtual server named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver?action=unset
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver+json
  - **Payload**

```
{
   "lbvserver":
   {
      "name":"MyFirstLbVServer",
      "lbmethod":true,
      "comment":true,
   }
}
```

Bind/Unbind

To bind a resource to another, specify the name of the two resources and specify the weight for the binding.

To bind a service named "svc_prod" to a load balancing virtual server named "MyFirstLbVServer", by specifying a certain weight for the binding:

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver_service_binding/
- **HTTP Method.** PUT
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver_service_binding+json
  - **Payload**

```
{
   "lbvserver_service_binding":
   {
      "name":"MyFirstLbVServer",
      "servicename":"svc_prod",
      "weight":111,
   }
}
```

Note: To unbind, specify the arguments in the URL as follows:

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver_service_binding/MyFirstLbVServer?args=servicename:svc_prod
- **HTTP Method.** DELETE

Bulk operations

You can create, retrieve, update, and delete multiple resources simultaneously and thus minimize network traffic. For

example, you can add multiple load balancing virtual servers in the same operation. To perform a bulk operation, specify the required parameters in the same request payload.

To account for the failure of some operations within the bulk operation, NITRO allows you to configure one of the following behaviors:

- **Exit.** When the first error is encountered, the execution stops. The commands that were executed before the error are committed.
- **Rollback.** When the first error is encountered, the execution stops. The commands that were executed before the error are rolled back. Rollback is only supported for add and bind commands.
- **Continue.** All the commands in the list are executed even if some commands fail.

You must specify the behavior of the bulk operation in the request header using the X-NITRO-ONERROR parameter.

To add two load balancing virtual servers in one operation and continue if one command fails:

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver_list+json
    X-NITRO-ONERROR:continue
  - **Payload**

```
{
  "lbvserver":
  [
    {
      "name":"new_lbvserver1",
      "servicetype":"http"
    },

    {
      "name":"new_lbvserver2",
      "servicetype":"http"
    }
  ]
}
```

- **Response**
  - **Header**
    HTTP/1.0 207 Multi Status
  - **Payload**

```
{
  "errorcode":273,
  "message":"Resource already exists",
  "severity":"ERROR",
  "response":
  [
    {
      "errorcode": 0,
```



```
        "message": "Done",
        "severity": "NONE"
      },
      {
        "errorcode": 273,
        "message":"Resource already exists",
        "severity": "ERROR"
      }
    ]
}
```

Binding NetScaler Resources

Feb 06, 2013

NetScaler resources form relationships with each other through the process of binding. This is how services are associated with an lbvserver (by binding them to it), or how various policies are bound to an lbvserver. Each binding relationship is represented by its own object. A binding resource has properties representing the name of each NetScaler resource in the binding relationship. It can also have other properties related to that relationship (for example, the weight of the binding between an lbvserver resource and a service resource).

Note: Unlike for NetScaler entities, you use a PUT HTTP method, instead of POST, for adding new binding resources. For more information on the REST messages, see the Configuration node of the <NITRO_SDK_HOME>/index.html file.

To bind a service to a load balancing virtual server named "MyFirstLbVServer" and specify a weight for the binding:

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver_service_binding/MyFirstLbVServer?action=bind
- **HTTP Method.** PUT
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver_service_binding+json
  - **Payload**

```
{
  "lbvserver_service_binding":
  {
    "servicename":"svc_prod",
    "weight":20,
    "name":"MyFirstLbVServer"
  }
}
```

To retrieve list of all the services bound to a virtual server "lbv1":

- **URL.** http://10.102.29.60/nitro/v1/config/lbvserver_service_binding/lbv1?attrs=servicename
- **HTTP Method.** GET

For more information on retrieving information, see the "Retrieving properties of a resource" section in Configuring NetScaler Features.

Globally Bind Resources

Some NetScaler resources can be bound globally to affect the whole system. For example, if a compression policy is bound to an lbvserver, the policy affects only the traffic on that lbvserver. However, if bound globally, it can affect any traffic on the appliance, regardless of which virtual servers handle the traffic.

The names of NITRO resources that can be used to bind resources globally have the pattern <featurename>global_<resourcetype>_binding. For example, the object aaaglobal_preauthenticationpolicy_binding is used to bind preauthentication policies globally.

To bind the policy named preautpol1 globally at priority 200:

- **URL.** http://10.102.29.60/nitro/v1/config/aaaglobal_aaapreauthenticationpolicy_binding?action=bind

- **HTTP Method.** PUT
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.aaaglobal_aaapreauthenticationpolicy_binding+json
  - **Payload**

```
{
  "aaaglobal_aaapreauthenticationpolicy_binding":
  {
    "policy":"preautpol1",
    "priority":200
  }
}
```

Configuring a NetScaler Cluster

Nov 05, 2013

You can use NITRO to add or create and manage a NetScaler cluster.

Cluster Instance Operations

All operations on a cluster instance must be performed on the clusterinstance object.

To create a cluster instance with ID 1:

- **URL.** http://10.102.29.60/nitro/v1/config/clusterinstance/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.clusterinstance+json
  - **Payload**

```
{
  "clusterinstance":
  {
    "clid":1,
    "preemption":"ENABLED"
  }
}
```

Cluster Node Operations

All operations on a cluster node must be performed on the clusternode object.

To add a cluster node with NSIP address 10.102.29.60:

- **URL.** http://10.102.29.60/nitro/v1/config/clusternode/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.clusternode+json
  - **Payload**

```
{
  "clusternode":
  {
    "nodeid":1,
    "ipaddress":"10.102.29.60",
    "state":"ACTIVE",
    "backplane":"1/1/2"
  }
}
```

Add a Cluster IP Address

To define a cluster IP address, specify the required parameters in the nsip object.

To configure a cluster IP address on NetScaler appliance with IP address 10.102.29.60:

- **URL.** http://10.102.29.60/nitro/v1/config/nsip/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.nsip+json
  - **Payload**

```
{
  "nsip":
  {
    "ipaddress":"10.102.29.61",
    "netmask":"255.255.255.255",
    "type":"CLIP"
  }
}
```

Add a Spotted IP Address

To configure an IP address as spotted, specify the required parameters in the nsip object. This configuration must be done on the cluster IP address.

To configure a spotted SNIP address on a node with ID 1:

- **URL.** http://10.102.29.60/nitro/v1/config/nsip/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.nsip+json
  - **Payload**

```
{
  "nsip":
  {
    "ipaddress":"10.102.29.77",
    "netmask":"255.255.255.0",
    "type":"SNIP",
    "ownernode":1
  }
}
```

Join NetScaler Appliance to Cluster

To join an appliance to a cluster, specify the required parameters in the cluster object.

To join a NetScaler appliance to a cluster:

- **URL.** http://10.102.29.60/nitro/v1/config/cluster/
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.cluster+json
  - **Payload**

```
{
  "cluster":
  {
    "clip":"10.102.29.61",
    "password":"verysecret"
  }
}
```

Linkset Operations

To configure a linkset, do the following:

1. Create a linkset by specifying the required parameters in the linkset object.
   To add a linkset LS/1:
   - **URL.** http://10.102.29.60/nitro/v1/config/linkset/
   - **HTTP Method.** POST
   - **Request.**
     - **Header**
       Cookie:NITRO_AUTH_TOKEN=tokenvalue
       Content-Type:application/vnd.com.citrix.netscaler.linkset+json
     - **Payload**

```
{
  "linkset":
  {
    "id":"LS/1"
  }
}
```

2. Bind the required interfaces to the linkset by specifying the interfaces in the linkset_interface_binding object.
   To bind interfaces 1/1/2 and 2/1/2 to linkset LS/1:
   - **URL.** http://10.102.29.60/nitro/v1/config/linkset_interface_binding/ LS%2F1?action=bind
   - **HTTP Method.** PUT
   - **Request.**
     - **Header**
       Cookie:NITRO_AUTH_TOKEN=tokenvalue
       Content-Type:application/vnd.com.citrix.netscaler.linkset_interface_binding+json
     - **Payload**

```
{
  "linkset_interface_binding":
  {
```

```
      "id":"LS/1",
      "ifnum":"1/1/2 2/1/2"
    }
  }
```

Retrieving Feature Statistics

Feb 06, 2013

The NetScaler appliance collects statistics about the usage of its features and the corresponding resources. NITRO can retrieve these statistics.

- URL to get statistics of a feature must have the format http://<NSIP>/nitro/v1/stat/<feature_name>.
- URL to get the statistics of a resource must have the format: http://<NSIP>/nitro/v1/stat/<resource_type>/<resource_name>.

For more information on the REST messages, see the Statistics node of the <NITRO_SDK_HOME>/index.html file.

To get the statistics of a lbvserver named "MyFirstLbVServer":

- **URL.** http://10.102.29.60/nitro/v1/stat/lbvserver/MyFirstLbVServer
- **HTTP Method.** GET
- **Request.**
  - **Header.**
    Content-Type:application/vnd.com.citrix.netscaler.lbvserver+json
- **Response.**
  - **Header**
    HTTP/1.0 200 OK
  - **Payload**

```
{
   "lbvserver":
   [
      {
         "name":"MyFirstLbVServer",
         "establishedconn":0,
         "vslbhealth":0,
         "primaryipaddress":"0.0.0.0",
         ...
      }
   ]
}
```

Note: Not all NetScaler features and resources have statistic objects associated with them.

Managing AppExpert Applications

Feb 06, 2013

To export an AppExpert application, specify the parameters needed for the export operation in the apptemplateinfo object. Optionally, you can specify basic information about the AppExpert application template, such as the author of the configuration, a summary of the template functionality, and the template version number, in the template_info object. This information is stored as part of the template file that is created.

To export an AppExpert application named "MyApp1":

- **URL.** http://10.102.29.60/nitro/v1/config/apptemplateinfo?action=export
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.apptemplateinfo+json
  - **Payload**

```
{
  "apptemplateinfo":
  {
    "appname":"MyApp1",
    "apptemplatefilename":"BizAp.xml",
    "template_info":
    {
      "templateversion_major":"2",
      "templateversion_minor":"1",
      "author":"XYZ",
      "introduction":"Intro",
      "summary":"Summary"
    },
  }
}
```

To import an AppExpert application, specify the parameters needed for the import operation in the apptemplateinfo object.

To import an AppExpert application named "MyApp1":

- **URL.** http://10.102.29.60/nitro/v1/config/apptemplateinfo?action=import
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.apptemplateinfo+json
    X-NITRO-ONERROR:rollback
  - **Payload**

```
{
  "apptemplateinfo":
```

```
    {
      "apptemplatefilename":"BizAp.xml",
      "deploymentfilename":"BizAp_deployment.xml",
      "appname":"MyApp1"
    }
  }
```

To import an AppExpert application by specifying different deployment settings:

- **URL.** http://10.102.29.60/nitro/v1/config/apptemplateinfo?action=import
- **HTTP Method.** POST
- **Request.**
  - **Header**
    Cookie:NITRO_AUTH_TOKEN=tokenvalue
    Content-Type:application/vnd.com.citrix.netscaler.apptemplateinfo+json
    X-NITRO-ONERROR:rollback
  - **Payload**

```
{
  "apptemplateinfo":
  {
    "apptemplatefilename":"BizAp.xml",
    "appname":"Myapp2"
     "deploymentinfo":
     {
       "appendpoint":
       [
         {
           "ipv46":"11.2.3.8",
           "port":80,
           "servicetype":"HTTP"
         }
       ],
       "service":
       [
         {
           "ip":"12.3.3.15",
           "port":80,
           "servicetype":"SSL"
         },
         {
           "ip":"14.5.5.16",
           "port":443,
           "servicetype":"SSL"
         }
       ],
    }
  }
}
```

Handling Exceptions

Feb 06, 2013

The response header provides the status of an operation by using HTTP status codes and the response payload provides the requested resource object (for GET method) and error details (for unsuccessful operation). NITRO does not provide a response payload for successful POST, PUT and DELETE methods. For successful GET method, the response payload consists only the requested resource object.

The following table provides the HTTP status codes:

Status	HTTP Status Code	Description
Success	200 OK	Request successfully executed.
	201 CREATED	Entity created.
Failure	400 Bad Request	Incorrect request provided.
	401 unauthorized	Not provided login credentials.
	403 forbidden	User is unauthorized
	404 Not Found	User is trying to access a resource not present in the NetScaler.
	405 Method Not Allowed	User is trying to access request methods not supported by NITRO.
	406 Not Acceptable	None of the values supplied by the user in the Accept header can be satisfied by the server.
	409 Conflict	The resource already exists on the NetScaler.
	503 Service Unavailable	The service is not available.
	599	NetScaler specific error code.
Warning	209 X-NITRO-WARNING	Warnings are captured by specifying the login URL as http://<nsip>/nitro/v1/config/login/?warning=yes.

Status	HTTP Status Code	Description
Combination of success and failure (for bulk operation with X-NITRO-ONERROR set as continue)	207 Multi	Some commands are executed successfully and some have failed.

Note: The content-type in the response header of an unsuccessful operation, consists of error MIME type instead of resource MIME type.
For a more detailed description of the error codes, see the API reference available in the <NITRO_SDK_HOME>/doc folder.

NITRO Changes Across NetScaler Releases

Jul 04, 2014

NetScaler has introduced some changes in the NITRO API since the NetScaler 9.3 release. This could raise some compatibility issues for the following users:

- Users migrating from NetScaler 9.3 to 10.1
- Users migrating from NetScaler 9.3 to 10.5

Note: There are no changes introduced since the NetScaler 10.1 release. Therefore, you should not face any compatibility issues when migrating from NetScaler 10.1 to 10.5.

These NITRO changes from 9.3 to 10.1 or 10.5 are categorized as follows:

- Resources Removed
- APIs Removed
- API Return Type Changed
- Attribute Type Changed
- Attributes Removed
- SDK Specific Changes

Note: Unless otherwise specified, these changes are applicable to both REST and SDKs.

Resource	Replace with...	Comments
lbmonitor_lbmetrictable_binding	lbmonitor_metric_binding	

Resource	API	Comments
vserver	GET	Perform the GET operation on specific virtual server types such as lb/cr/cs.
filterpolicy	POST with "action=unset"	This API is removed as unsetting the attributes('action') of a policy makes it invalid.
auditsyslogpolicy	POST with "action=unset"	This API is removed as unsetting the attributes('action') of a policy makes it invalid.
auditnslogpolicy	POST with "action=unset"	This API is removed as unsetting the attributes('action') of a policy makes it invalid.
authorizationpolicy	POST with "action=unset"	This API is removed as unsetting the attributes('action') of a policy makes it invalid.

Resource	API	Comments
snmpengineid	GET	Return type changed to an array.
nshostname	GET	Return type changed to an array.

Resource	Attribute	Comments
appfwpolicy_lbvserver_binding	activepolicy	Data type changed from Boolean to Integer.
appfwpolicy_appfwglobal_binding	activepolicy	Data type changed from Boolean to Integer.
vlan	portbitmap	Data type changed from uint to ulong.
vlan	tagbitmap	Data type changed from uint to ulong.

Resource	Attribute	Replace with...	Comments
policypatset_pattern_binding	indextype	- NA -	This attribute is moved to 'policypatset' resource as this attribute is applicable at patset level.
system_stats	powersupply1failure	powersupply1status	Change is applicable from NetScaler 9.3 (65.8).
system_stats	powersupply2failure	powersupply2status	Change is applicable from NetScaler 9.3 (65.8).
server_servicegroup_binding	servicetype	svctype	
server_service_binding	servicetype	svctype	
crvserver	hits	- NA -	Hits are calculated per policy binding hence moved this parameter to binding resources.
crvserver	dstvsvr	destinationvserver	
crvserver	destvserver	domain	

Resource	Attribute	Replace with...	Comments
crvserver	dnsvserver	dnsvservername	
appflowpolicylabel	type	policylabeltype	
sslcipher	ciphgrpals	ciphergroupname	This change is applicable for sslcipher_*_binding resources also.
csvserver_cspolicy_binding	targetvserver	targetlbvserver	
csvserver_cspolicy_binding	targetvserver	targetlbvserver	
rewriteaction	allow_unsafe_pi1, allow_unsafe_pi	bypassSafetyCheck	

Class	Method	Replace with...	Comments
Routerbgp	- NA -	- NA -	This class is removed as all router configurations are deprecated in 9.2.
dnsptrrec	get(dnsptrrec obj, nitro_service session)	get(nitro_service session, String reversedomain)	
dnsaddrec	get(dnsaddrec obj, nitro_service session)	get(nitro_service session, String hostname)	
dnsnsrec	get(dnsnsrec obj, nitro_service session)	get(nitro_service session, String domain)	
snmpengineid	unset(nitro_service session, String[] args)	unset(nitro_service session, snmpengineid resource, String[] args)	
arp	arp.get(nitro_service session, String ipaddress)	arp.get(nitro_service session, arp resource)	
nsip	get(nitro_service session, String ipaddress)	get(nitro_service client, nsip resource)	
nsip6	get(nitro_service session, String ipv6address)	get(nitro_service session, nsip6 resource)	
dnsmxrec	dnsmxrec.get(dnsmxrec obj, nitro_service session)	dnsmxrec[] get(nitro_service service, dnsmxrec_args args)	

Unsupported NetScaler Operations

Jul 10, 2013

Some NetScaler operations that are available through the command line interface and through the configuration utility, are not available through NITRO APIs. The following list provides the NetScaler operations not supported by NITRO:

- install API
- diff API on nsconfig resource
- UI-internal APIs (update, unset, and get)
- show ns info
- Application firewall APIs:
  - importwsdl
  - importcustom
  - importxmlschema
  - importxmlerrorpage
  - importhtmlerrorpage
  - rmwsdl
  - rmcustom
  - rmxmlschema
  - rmxmlerrorpage
  - rmhtmlerrorpage
- CLI-specific APIs:
  - ping
  - ping6
  - traceroute
  - traceroute6
  - nstrace
  - scp
  - configaudit
  - show defaults
  - show permission
  - batch
  - source

XML API

Mar 19, 2012

Developers and administrators can use the NetScaler Application Programming Interface (API), nsconfig, to implement customized client applications. The nsconfig API, which mirrors the NetScaler command line interface (CLI), is based on the Web Services Description (WSDL) specification. It includes a filterwsdl command to reduce compilation time and file size. You can secure your API applications at the NetScaler IP address or at the IP address of the subnet on which the NetScaler is deployed.

The following topics describe the properties and use of the API.

Introduction	General information about the API, requirements, and software-version information.
The NS Config Interface	How to use the API.
Examples of API Usage	Basic examples of how to use the API.
The Web Service Description Language (WSDL)	How to use the WSDL-based interface schema to support your client applications, and how to use WSDL Filter to reduce file size and compilation time.
Securing API Access	How to secure API access.

Introduction to the API

Mar 19, 2012

The API enables programmatic communications between client applications and the NetScaler appliance, providing the following benefits:

- Developers can control the NetScaler from a custom application. The API enables the client application to configure and monitor the NetScaler.
- Developers can create client applications easily and quickly, using a language and platform with which they are comfortable.
- The API provides a secure, end-to-end, standards-based framework that integrates into the existing infrastructure.

Based on the Simple Object Access Protocol (SOAP) over HTTP, the API consists of the NSConfig interface. NSConfig includes methods for setting and querying the configuration. These methods allow the client application using the NSConfig interface to perform almost all operations that an administrator would normally perform with the CLI or GUI.

In addition, the NetScaler provides an interface description, based on the Web Services Definition Language (WSDL), that facilitates the development of client applications.

Hardware and Software Requirements

Mar 19, 2012

To work with the API, your system needs to meet the following hardware and software setup and requirements:

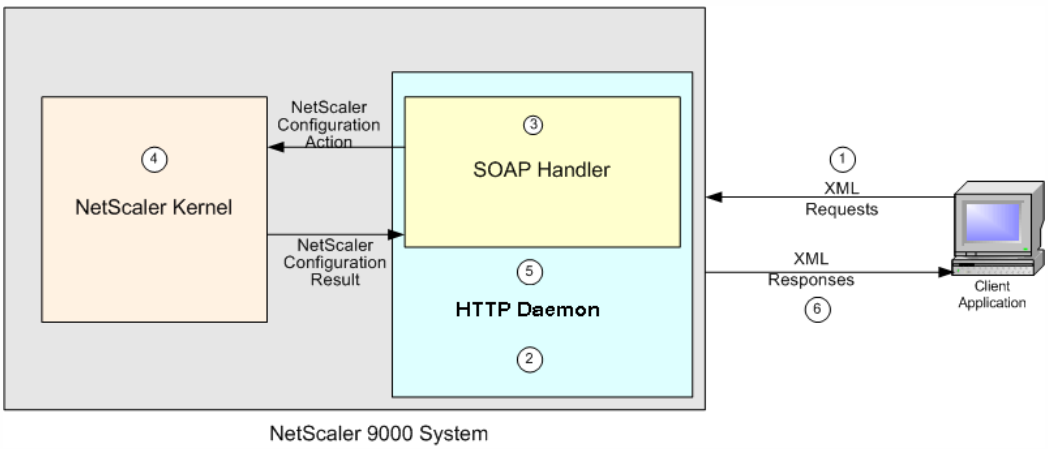
- A client workstation.
- Access to a NetScaler, version 8.0 or higher.
- A SOAP client tool kit (supporting SOAP version 1.1 and above), and the development environment for the tool kit. For example, if you use a Visual Basic tool kit, you must have Visual Basic installed on your system.

API Architecture

Mar 19, 2012

The API architecture is designed to allow NSConfig client requests to be routed, through the HTTP daemon running on the target NetScaler, to a SOAP handler that translates the SOAP request into a call to the (internal) kernel configuration API.

Figure 1. The API Architecture



The order in which the NetScaler processes requests through the API is as follows:

- The client formats a request containing XML conforming to the SOAP protocol and sends it to the NetScaler.
- The HTTPD server instance on the NetScaler routes this request to a SOAP handler.
- The SOAP handler interprets the SOAP headers and maps the enclosed request to an internal configuration function.
- The kernel acts on the request and returns one or more responses.
- The SOAP handler translates the response(s) to a SOAP response message.
- The XML response is sent back to the client in an HTTP response.

The NSConfig Interface

Mar 19, 2012

The NSConfig interface closely mirrors the structure of the NetScaler command line interface (CLI). Administrators and programmers who are familiar with the CLI can easily create and implement custom applications to query or set the configuration on their NetScaler.

The NSConfig interface includes methods for most of the CLI commands. In most cases the method and the command name are the same. See the PortType section of the WSDL for a complete list of methods and their names.

For example, you use the add lb vserver CLI command to create a load balancing virtual server, as follows:

add lb vserver <vServerName> <serviceType> [<IPAddress> <port>]

where:

<vServerName> = A name for the virtual server.

<serviceType> = (HTTP | FTP | TCP | UDP).

<IPAddress> = The IP address used by the virtual server.

<port> = The port that the virtual server listens on.

Following is the corresponding API call, in the C language:

```
int ns__addlbvserver(void *handle,
  string vServerName,
  string serviceType,
  string IPAddress,
  unsignedShort port,
  ns__addlbvserverResponse *out);
```

Note: The exact syntax of the API call depends on the language used to write the client program. The above `ns__addlbvserver` function prototype is similar to the one that would be generated by the gSOAP package at http://www.cs.fsu.edu/~engelen/soap.html.

The result returned for all NSConfig requests consists of:

Rc

An integer return code. The value is zero if the request succeeded. A non-zero value indicates that the request failed.

Message

A string message. Contains meaningful information only if the request fails (rc is non-zero) (for example, “Required argument missing”).

List

A type-specific list of result entities. This element is present only for requests that retrieve information from the NetScaler. For example, the API method names starting with get, which correspond to the CLI show commands, return a list.

Command names in the NetScaler CLI typically consist of three terms, separated by spaces, identifying the operation, the feature that is being operated on, and the specific item that is being operated on. For example, to create a new application firewall profile, you type add appfw profile, followed by the command arguments. The corresponding API methods omit the spaces. For example, the API method for add appfw profile is `addappfwprofile`. The same principle applies to CLI

command names that have only two terms. For example, add monitor becomes addmonitor. The other exceptions to this pattern are as follows:

1. The CLI show command is changed to get in the API, as shown below.
   show lb vserver => getlbvserver

   show service => getservice

2. The following commands are omitted from the API:
   - Commands that apply to the CLI itself (for example, clear CLI prompt).
   - The batch, ping, grep, more, shell, and scp commands.
   - The show router bgp and show router map commands.
   - All stat commands.
3. Message "part" names in the API are the same as the corresponding CLI argument names. As in the CLI, case does not matter, and these names can be abbreviated. For more information, *Citrix NetScaler Command Reference Guide* at
4. The result of a GET method (which corresponds to a show command in the CLI) is always an array of a type defined in the WSDL. The elements of these complex types generally correspond to arguments to the corresponding add/set command/method.
5. Authorization must be performed once, by sending a login request. The response contains a Set-Cookie HTTP header, and the cookie must be sent with each subsequent request. This is addressed in the Perl examples using by HTTP::Cookies. HTTP::Cookies are used for API client authentication purposes (to log into the NetScaler). In Perl, SOAP::Lite cannot perform this authentication process; HTTP::Cookies are used instead.
6. In some programming languages, such as Perl, it is possible to invoke the programming language API without using the WSDL.

Examples of API Usage

Mar 19, 2012

The following examples show how to develop an API call from a standard CLI command, how to generate the SOAP request, and how the NetScaler responds to that request:

Example: Setting the Configuration

Example: Querying the Configuration

Example: Setting the Configuration

Mar 19, 2012

This example shows a CLI command, the corresponding API method, the resulting XML request, and the XML response that is sent back to the client.

Note: The actual API method and the XML SOAP message contents may differ from the example shown below. The XML shown will be encased in a SOAP envelope, which will in turn be carried in an HTTP message. For more information, see the W3C web site at http://www.w3.org/TR/SOAP.

The following CLI command creates a Load Balancing virtual server:

> add lb vserver vipLB1 HTTP 10.100.101.1 80

Following is the corresponding API method:

> ns__addlbvserver (handle, “vipLB1”, “HTTP”, “10.100.101.1”, 80, &out);

The XML generated for this request is as follows.

```
<ns:addlbvserver>
<vServerName xsi:type="xsd:string" >vipLB1</vServerName>
<serviceType xsi:type="ns:vservicetypeEnum>HTTP</ serviceType>
<IPAddress xsi:type="xsd:string">10.100.101.1</IPAddress>
<port xsi:type="xsd:unsignedInt" >80</port>
< /ns:addlbvserver >
```

The XML response to the above request is as follows.

```
<ns:addlbvserverResponse>
<rc xsi:type="xsd:unsignedInt">0</rc>
<message xsi:type="xsd:string">Done</message>
</ns:addlbvserverResponse>
```

Example: Querying the Configuration

Mar 19, 2012

This example shows an API request that queries the configuration and receives a list of entities.

Note: The actual API method and the XML SOAP message contents may differ from the example shown below.
The following CLI command shows the configured Load Balancing virtual servers:

> show lb vservers

Sample output of the show lb vservers command is as follows.

```
> show lb vservers
2 configured virtual servers:
1)  vipLB1 (10.100.101.1:80) - HTTP Type: ADDRESS State:
   DOWN
   Method: LEASTCONNECTION Mode: IP
   Persistence: NONE
2)  vipLB2 (10.100.101.2:80) - HTTP Type: ADDRESS State:
   DOWN
   Method: LEASTCONNECTION Mode: IP
   Persistence: NONE
Done
```

Following is the corresponding API method to show the list of Load Balancing virtual servers.

ns__getlbvserver(handle, NULL, &out)

The XML generated for this request is as follows.

```
<ns:getlbvserver></ns:getlbvserver>
```

The XML response to the above request is as follows.

```
<ns:getlbvserverResponse>
 <rc xsi:type="xsd:unsignedInt">0</rc>
 <message xsi:type="xsd:string">Done</message>
 <List xsi:type="SOAP-ENC:Array"
  SOAP-ENC:arrayType="ns:lbvserver[2]">
  <item xsi:type="ns:lbvserver">
  <vServerName xsi:type="xsd:string>vipLB1
   </vServerName>
   <serviceType xsi:type="xsd:string>HTTP</ serviceType>
   <IPAddress xsi:type="xsd:string >10.100.101.1
   </IPAddress>
  <port xsi:type="xsd:unsignedInt">80</port>
 </item>
 <item xsi:type="ns:lbvserver">
  <vServerName xsi:type="xsd:string>vipLB2
  </vServerName>
  <serviceType xsi:type="xsd:string>HTTP</ serviceType>
```

```
  <IPAddress xsi:type="xsd:string >10.100.101.2
  </IPAddress>
   <port xsi:type="xsd:unsignedInt">80</port>
  </item>
 </List>
</ns:getlbvserverResponse>
```

The Web Service Definition Language (WSDL)

Mar 19, 2012

The NetScaler WSDL describes services for the entire range of NetScaler services. The NetScaler provides two WSDL files:

NSConfig.wsdl

Configuration APIs are defined in this file. The NSConfig.wsdl file is found on the NetScaler at http://<NSIP>/api/NSConfig.wsdl, where <NSIP> is the IP address of your NetScaler. This file is much larger than the NSStat.wsdl file. With the help of a third-party tool (such as gSOAP), developers can use this file to generate client stubs. A custom application can then call the stubs to send requests to the NetScaler. The application can be in any standard programming language that is supported by the third-party tool. Common programming languages for this purpose include Perl, Java, C, and C#. You can use the filterwsdl command to select only the service definitions that are relevant to the API calls made in your script.

NSStat.wsdl

Statistical APIs are defined in this file. The NSStat.wsdl file is found on the NetScaler at http://<NSIP>/api/NSStat.wsdl, where <NSIP> is the IP address of your NetScaler.

Creating Client Applications with the NSConfig.wsdl File

Mar 19, 2012

A client application can be created by importing the NSConfig.wsdl file with the gSOAP WSDL Importer to create a header file with C or C++ declarations of the SOAP methods. The gSOAP compiler is then used to translate this header file into stubs for the client application.

1. Get the NSConfig.h header file from the WSDL file.
   1. Run the wsdl2h program that comes with gSOAP on the WSDL file. The wsdl2h program is in the following location.
      > ./wsdl2h NSConfig.wsdl

      The output of wsdl2h is as follows:
      ```
      ** The gSOAP WSDL parser for C and C++ 1.0.2
      ** Copyright (C) 2001-2004 Robert van Engelen, Genivia, Inc.
      ** All Rights Reserved. This product is provided "as is", without any warranty.
      Saving NSConfig.h
      Reading file 'NSConfig.wsdl'
      Cannot open file 'typemap.dat'
      Problem reading type map file typemap.dat.
      Using internal type definitions for C instead.
      ```
   2. Run the soapcpp2 program to compile the header file and complete the process, as shown below. > soapcpp2 NSConfig.h
2. Generate the XML files and stubs as follows:
   > ./soapcpp2 -c -i NSConfig.h

   Following is sample output for this command:

   ```
   ** The gSOAP Stub and Skeleton Compiler for C and C++ 2.4.1
   ** Copyright (C) 2001-2004 Robert van Engelen, Genivia, Inc.
   ** All Rights Reserved. This product is provided "as is", without any warranty.
   Saving soapStub.h
   Saving soapH.h
   Saving soapC.c
   Saving soapClient.c
   Saving soapServer.c
   Saving soapClientLib.c
   Saving soapServerLib.c
   Using ns1 service name: NSConfigBinding
   Using ns1 service location: http://NetScaler.com/api Using ns1 schema namespace: urn:NSConfig
   Saving soapNSConfigBindingProxy.h client proxy
   Saving soapNSConfigBindingObject.h server object
   Saving NSConfigBinding.addserver.req.xml sample SOAP/XML request
   Saving NSConfigBinding.addserver.res.xml sample SOAP/XML response
   Saving NSConfigBinding.disableserver.req.xml sample SOAP/ XML request
   Saving NSConfigBinding.disableserver.res.xml sample SOAP/ XML response
   ```

```
Saving NSConfigBinding.enableserver.req.xml sample SOAP/XML request
Saving NSConfigBinding.enableserver.res.xml sample SOAP/XML response
[ ... Similar lines clipped ... ]
Saving NSConfigBinding.nsmap namespace mapping table
Compilation successful
```

This creates the stub files soapC.c, soapClient.c and stdsoap2.c.

3. Link the stub files you created with your source code to create a stand-alone binary that invokes the API.

Filter WSDL

Mar 19, 2012

The NetScaler WSDL describes services for the entire range of NetScaler services. When you use the NetScaler API in your scripts, by linking to the WSDL and attempting to compile the application, the entire WSDL is included, unnecessarily increasing compilation time and the size of the program.

Filter WSDL is a tool for selecting only those service definitions from the NetScaler WSDL that are relevant to the API calls made in the script. You can use the filter WSDL tool to filter NSConfig.wsdl and NSStat.wsdl files.

The NetScaler provides two WSDL files, one for the configuration APIs (NSConfig.wsdl) and the other for statistical APIs (NSStat.wsdl). The WSDL file for the configuration API is much larger. Therefore, it is important to use filter WSDL when compiling programs written with the configuration API.

Filter WSDL is a program that works on the Windows, FreeBSD and Linux platforms, and it can be run from the CLI.

The syntax for running filter WSDL is as follows:

filterwsdl <fromwsdl> <pattern>

where:

fromwsdl = The wsdl file that you want to filter

pattern = API method names or patterns that should be filtered

For example, if you want to filter all the service definitions for the API method addlbvserver from the NetScaler WSDL file, NSConfig.wsdl, you can use the command:

> filterwsdl NSConfig.wsdl "addlbvserver"

The output of this command is sent to the screen by default, but it can be redirected to a file on the NetScaler by using the UNIX redirect operator (>). The output of the previous command can be saved into a file called NSConfig-Custom.wsdl by using the command as follows:

> filterwsdl NSConfig.wsdl "addlbvserver" > NSConfig-Custom.wsdl

In this case, the original WSDL file is 1.58 MB, but the filtered WSDL file is 6 KB.

The pattern used in the filterwsdl command can include the + and - operators and the wildcard operator (*) to create more generic filters.

For example, if you want to filter the service definitions for all the available load balancing methods, you can use the following command:

> filterwsdl NSConfig.wsdl "*lb"*

This command will filter all the Load Balancing methods but will also include GSLB methods, because the pattern lb will be matched by all GSLB methods also. To include only LB methods and exclude all GSLB methods, use the command as follows:

> filterwsdl NSConfig.wsdl +"*lb" -"glsb"*

Securing API Access

Mar 19, 2012

Secure access to CLI objects can be based on the NetScaler IP address or on the subnet IP address on which the NetScaler is deployed. To provide secured API access based on the NetScaler IP address, you must configure the NetScaler to use transparent SSL mode with clear text port.

1. Create a loopback SSL service and configure it use transparent SSL mode with clear text port:
   add service secure_xmlaccess 127.0.0.1 SSL 443 -clearTextPort 80
2. Add certificate and key:
   add certkey cert1 –cert /nsconfig/ssl/ssl/cert1024.pem –key /nsconfig/ssl/ssl/rsakey.pem

   Note: You can use an existing certificate and key or use the NetScaler Certificate Authority Tool to create a key and test certificate for secure access.
3. Bind the certificate and key to the service:
   bind certkey secure_xmlaccess cert1 -Service
4. Add a custom TCP monitor to monitor the SSL service you have added:
   add monitor ssl_mon TCP -destport 80
5. Bind the custom TCP monitor to the SSL service:
   bind monitor ssl_mon secure_xmlaccess

1. Create an SSL VIP in the appropriate subnet:
   add vserver <vServerName> SSL <Subnet-IP> 443
2. Create a loopback HTTP service:
   add service <serviceName> 127.0.0.1 HTTP 80
3. Bind the service to the SSL VIP:
   bind lb vserver <vServerName> <serviceName>
4. Add the certificate and the key:
   add certkey cert1 –cert /nsconfig/ssl/ssl/cert1024.pem –key /nsconfig/ssl/ssl/rsakey.pem

   Note: You can use an existing certificate and key or use the NetScaler Certificate Authority Tool to create a key and test certificate.
5. Bind the Certificate and the Key to the SSL VIP:
   bind certkey <vServerName> cert1

Reference Material

Sep 22, 2015

Use the reference information in this section to get an in-depth understanding of the following NetScaler components:

- NetScaler SNMP OIDs - Details of the SNMP OIDs that can be used to obtain information from a NetScaler appliance.
- NetScaler Syslog Messages - Details of the Syslog messages given by the NetScaler appliance.
- NetScaler CLI Commands - Details of the commands that can be used to configure the NetScaler appliance through the CLI. You can also view the details of each command in the NetScaler CLI, by entering the "man <ns-command-name>" command.
- Policy Expressions - Details of the policy expressions available on the NetScaler.
- Quick Start Guides - A reference to quick installation and configuration of your hardware appliance.

Internal Server Error

The server encountered an internal error or misconfiguration and was unable to complete your request.

Please contact the server administrator, webmaster@localhost and inform them of the time the error occurred, and anything you might have done that may have caused the error.

More information about this error may be available in the server error log.

Apache/2.2.31 (Amazon) Server at 10.57.13.146 Port 80

NetScaler Command Reference

Sep 22, 2015

A detailed list of the commands that can be used to configure the NetScaler appliance through the CLI.

- Traffic Management Commands
- Transform Commands
- Tunnel Commands
- Utility Commands
- VPN Commands
- WebInterface Commands

AAA Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- aaa
- aaa certParams
- aaa global
- aaa group
- aaa kcdAccount
- aaa ldapParams
- aaa parameter
- aaa preauthenticationaction
- aaa preauthenticationparameter
- aaa preauthenticationpolicy
- aaa radiusParams
- aaa session
- aaa stats
- aaa tacacsParams
- aaa user

aaa

Sep 22, 2015

The following operations can be performed on "aaa":

Display aaa statistics

stat aaa [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Authentication successes (authsucc)

Count of authentication successes.

Authentication failures (authfails)

Count of authentication failures.

HTTP authorization successes (atzhtps)

Count of HTTP connections that succeeded authorization.

HTTP authorization failures (atzhtpf)

Count of HTTP connections that failed authorization.

Non HTTP authorization successes (atznonhtps)

Count of non HTTP connections that succeeded authorization.

Non HTTP authorization failures (atznonhtpf)

Count of non HTTP connections that failed authorization.

Current AAA sessions (totcursess)

Count of current AAA sessions.

Total AAA sessions (totsess)

Count of all AAA sessions.

Timed out AAA sessions (totsessto)

Count of AAA sessions that have timed out.

Current ICAOnly sessions (totcuricasess)

Count of current ICA only sessions.

Current ICAOnly Conn (curicaonlyconn)

Count of current ICA only connections.

Current ICA (Smart Access) Conn (curicaconn)

Count of current ICA connections.

Current TM sessions (curTMses)

Count of current AAATM sessions.

TM sessions (totTMses)

Count of all AAATM sessions.

aaa certParams

Sep 22, 2015

The following operations can be performed on "aaa certParams":

set | unset | show

set aaa certParams

Modifies the global configuration settings for certificate policies. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

set aaa certParams [-userNameField <string>] [-groupNameField <string>] [-defaultAuthenticationGroup <string>]

Arguments

userNameField

Client certificate field that contains the username, in the format <field>:<subfield>.

groupNameField

Client certificate field that specifies the group, in the format <field>:<subfield>.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

To configure the default certificate parameters: set aaa certparams -userNameField "Subject:CN" -groupNameField "Subject:OU"

---

unset aaa certParams

Use this command to remove aaa certParams settings.Refer to the set aaa certParams command for meanings of the arguments.

Synopsys

unset aaa certParams [-userNameField] [-groupNameField] [-defaultAuthenticationGroup]

---

show aaa certParams

Displays the current client certificate configuration on the NetScaler appliance.

Synopsys

show aaa certParams

Arguments

format

level

Outputs

twoFactor

The state of the two-factor authentication.

userNameField

The field in the certificate from which the username will be extracted.

groupNameField

The field in the certificate from which the group will be extracted.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

aaa global

Sep 22, 2015

The following operations can be performed on "aaa global":

bind | unbind | show

bind aaa global

Binds a policy globally.

Synopsys

bind aaa global [-policy <string> [-priority <positive_integer>]] [-windowsProfile <string>]

Arguments

policy

Name of the policy to bind globally.

windowsProfile

Name of the negotiate profile to bind globally.

Example

```
bind aaa global -pol pol1
```

---

unbind aaa global

Unbind the policy from the global bind point.

Synopsys

unbind aaa global [-policy <string>] [-windowsProfile <string>]

Arguments

policy

Name of the policy to be unbound.

windowsProfile

Name of the negotiate profile to be bound.

---

show aaa global

Displays a list of policies that are currently bound to Global on the NetScaler appliance.

Synopsys

show aaa global

Arguments

summary

fullValues

format

level

Outputs

policy

Name of the policy to be unbound.

windowsProfile

Name of the negotiate profile to be bound.

priority

Priority of the bound policy

bindPolicyType

Bound policy type

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

aaa group

Sep 22, 2015

The following operations can be performed on "aaa group":

add | rm | bind | unbind | show

add aaa group

Creates a AAA group and verifies the configuration to ensure that it is correct.

Synopsys

add aaa group <groupName>

Arguments

groupName

Name for the group. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the group is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or

single quotation marks (for example, ?my aaa group? or ?my aaa

group).

Example

```
add aaa group group_ad
```

---

rm aaa group

Removes the specified AAA group.

Synopsys

rm aaa group <groupName>

Arguments

groupName

Name of the group that you are removing.

---

bind aaa group

Binds the specified AAA group to the specified resource. The resource can be a user, an Intranet IP address or range, a policy, or an Intranet application.

Synopsys

bind aaa group <groupName> [-userName <string>] [-policy <string> [-priority <positive_integer>]] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>]

Arguments

groupName

Name of the group that you are binding.

userName

Bind a AAA group to the specified AAA user.

If the specified user is bound to more than one group, the group expressions are evaluated, upon authorization, to determine the appropriate action.

policy

Bind a policy to the specified AAA group.

intranetApplication

Bind the group to the specified intranet VPN application.

urlName

Bind the group to the specified URL.

intranetIP

Bind the group to the specified IP address or IP block.

Normally you would bind the group to an IP address or range that your users use to access intranet resources.

Example

To bind an Intranet IP to the group engg: bind aaa group engg -intranetip 10.102.10.0 255.255.255.0

---

unbind aaa group

Unbinds the specified AAA group from the specified resource. The resource can be a user, an intranet IP address or range, a policy, or an intranet application.

Synopsys

unbind aaa group <groupName> [-userName <string> ...] [-policy <string>] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>]

Arguments

groupName

Name of the group that you are unbinding.

userName

Unbind the specified AAA group from the specified AAA user.

policy

Unbind the specified policy from the specified AAA group.

intranetApplication

Unbind the specified group from the specified intranet VPN application.

urlName

Unbind the specified group from the specified URL.

intranetIP

Unbind the specified group from the specified IP address or IP block.

Example

unbind aaa group engg -intranetip 10.102.10.0 255.255.255.0

---

show aaa group

Displays the current configuration of a AAA group.

Synopsys

show aaa group [<groupName>] [-loggedIn]

Arguments

groupName

Name of the group.

loggedIn

Display only the group members who are currently logged in.

summary

fullValues

format

level

Outputs

userName

The user name.

policy

The policy name.

priority

Priority to assign to the policy, as an integer. A lower number indicates a higher priority.

Required when binding a group to a policy. Not relevant to any other

type of group binding.

intranetApplication

Bind the group to the specified intranet VPN application.

urlName

The intranet url

actType

intranetIP

The Intranet IP(s) bound to the group

netmask

The netmask for the Intranet IP

policySubType

stateflag

devno

count

Example

> show aaa group engg GroupName: engg Bound AAA users: UserName: joe UserName: jane In

aaa kcdAccount

Sep 22, 2015

The following operations can be performed on "aaa kcdAccount":

add | rm | set | unset | show

add aaa kcdAccount

Add a Kerberos constrained delegation account.

Synopsys

add aaa kcdAccount <kcdAccount> {-keytab <string>} {-realmStr <string>} {-delegatedUser <string>} {-kcdPassword } {-usercert <string>} {-cacert <string>} [-userRealm <string>]

Arguments

kcdAccount

The name of the KCD account.

keytab

The path to the keytab file. If specified other parameters in this command need not be given

realmStr

Kerberos Realm.

delegatedUser

Username that can perform kerberos constrained delegation.

kcdPassword

Password for Delegated User.

usercert

SSL Cert (including private key) for Delegated User.

cacert

CA Cert for UserCert or when doing PKINIT backchannel.

userRealm

Realm of the user

Example

add aaa kcdaccount my_kcd_acct -keytab /var/mykcd.keytab add aaa kcdaccount my_kcd_acct -keytab The above example adds a Kerberos constrained delegation account

---

rm aaa kcdAccount

Remove the KCD account.

Synopsys

rm aaa kcdAccount <kcdAccount>

Arguments

kcdAccount

The KCD account name.

---

set aaa kcdAccount

Set the KCD account information.

Synopsys

set aaa kcdAccount <kcdAccount> [-keytab <string>] [-realmStr <string>] [-delegatedUser <string>] [-kcdPassword] [-usercert <string>] [-cacert <string>] [-userRealm <string>]

Arguments

kcdAccount

The name of the KCD account.

keytab

The path to the keytab file. If specified other parameters in this command need not be given

realmStr

Kerberos Realm.

delegatedUser

Username that can perform kerberos constrained delegation.

kcdPassword

Password for Delegated User.

usercert

SSL Cert (including private key) for Delegated User.

cacert

CA Cert for UserCert or when doing PKINIT backchannel.

userRealm

Realm of the user

Example

set aaa kcdaccount my_kcd_acct -keytab /var/hiskcd.keytab The above command sets the keytab location for KCD account my_kcd_acct to /var/hiskcd.keytab

---

unset aaa kcdAccount

Unset the KCD account information..Refer to the set aaa kcdAccount command for meanings of the arguments.

Synopsys

unset aaa kcdAccount <kcdAccount> [-usercert <string>] [-cacert <string>] [-userRealm]

---

show aaa kcdAccount

Display KCD accounts.

Synopsys

show aaa kcdAccount [<kcdAccount>]

Arguments

kcdAccount

The KCD account name.

summary

fullValues

format

level

Outputs

keytab

The path to the keytab file. If specified other parameters in this command need not be given

principle

SPN extracted from keytab file.NOTE: This attribute is deprecated.This attribute is deprecated. Please do not configure this

kcdSPN

Host SPN extracted from keytab file.

realmStr

Kerberos Realm.

delegatedUser

Username that can perform kerberos constrained delegation.

kcdPassword

Password for Delegated User.

usercert

SSL Cert (including private key) for Delegated User.

cacert

CA Cert for UserCert or when doing PKINIT backchannel.

userRealm

Realm of the user

stateflag

devno

count

Example

Example > show aaa kcdaccount my_kcd_acct KcdAccount: my_kcd_acct Keytab: /var/mykcd.keytab Done >

aaa ldapParams

Sep 22, 2015

The following operations can be performed on "aaa ldapParams":

set | unset | show

set aaa ldapParams

Modifies the global configuration settings for the LDAP server. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

set aaa ldapParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-ldapLoginName <string>] [-searchFilter <string>] [-groupAttrName <string>] [-subAttributeName <string>] [-secType <secType>] [-svrType (AD | NDS)] [-ssoNameAttribute <string>] [-passwdChange (ENABLED | DISABLED)] [-nestedGroupExtraction (ON | OFF)] [-maxNestingLevel <positive_integer>] [-groupNameIdentifier <string>] [-groupSearchAttribute <string> [-groupSearchSubAttribute <string>]] [-groupSearchFilter <string>] [-defaultAuthenticationGroup <string>]

Arguments

serverIP

IP address of your LDAP server.

serverPort

Port number on which the LDAP server listens for connections.

Default value: 389

Minimum value: 1

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the LDAP server.

Default value: 3

Minimum value: 1

ldapBase

Base (the server and location) from which LDAP search commands should start.

If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

ldapBindDn

Complete distinguished name (DN) string used for binding to the LDAP server.

ldapBindDnPassword

Password for binding to the LDAP server.

ldapLoginName

Name attribute that the NetScaler appliance uses to query the external LDAP server or an Active Directory.

searchFilter

String to be combined with the default LDAP user search string to form the value to use when executing an LDAP search.

For example, the following values:

vpnallowed=true,

ldaploginame=""samaccount""

when combined with the user-supplied username ""bob"", yield the following LDAP search string:

""(&(vpnallowed=true)(samaccount=bob)""

groupAttrName

Attribute name used for group extraction from the LDAP server.

subAttributeName

Subattribute name used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

Possible values: PLAINTEXT, TLS, SSL

Default value: AAA_LDAP_PLAINTEXT

svrType

The type of LDAP server.

Possible values: AD, NDS

Default value: AAA_LDAP_SERVER_TYPE_DEFAULT

ssoNameAttribute

Attribute used by the NetScaler appliance to query an external LDAP server or Active Directory for an alternative username.

This alternative username is then used for single sign-on (SSO).

passwdChange

Accept password change requests.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nestedGroupExtraction

Queries the external LDAP server to determine whether the specified group belongs to another group.

Possible values: ON, OFF

Default value: OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

To configure authentication in the LDAP server running at 192.40.1.2: set aaa ldapparams -serverip 192.40.1.2 -ldapbase "dc=netscaler,dc=com" -ldapBindDN "cn=Mar

---

unset aaa ldapParams

Use this command to remove aaa ldapParams settings.Refer to the set aaa ldapParams command for meanings of the arguments.

Synopsys

unset aaa ldapParams [-serverIP] [-serverPort] [-authTimeout] [-ldapBase] [-ldapBindDn] [-ldapBindDnPassword] [-ldapLoginName] [-searchFilter] [-groupAttrName] [-subAttributeName] [-secType] [-svrType] [-ssoNameAttribute] [-passwdChange] [-nestedGroupExtraction] [-maxNestingLevel] [-groupNameIdentifier] [-groupSearchAttribute] [-groupSearchSubAttribute] [-groupSearchFilter] [-defaultAuthenticationGroup]

---

show aaa ldapParams

Displays the current LDAP configuration on the NetScaler appliance.

Synopsys

show aaa ldapParams

Arguments

format

level

Outputs

serverIP

The IP address of the LDAP server.

serverPort

Port number on which the LDAP server listens for connections.

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the LDAP server.

ldapBindDn

The full distinguished name used to bind to the LDAP server.

ldapLoginName

The name attribute used by the system to query the external LDAP server, or an Active Directory.

ldapBase

The base or node where the ldapsearch should start. If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

secType

The communication type between the system and the LDAP server.

svrType

LDAP server.

ssoNameAttribute

The attribute used by the system to query the external LDAP server, or an Active Directory, for an alternate username to be used in Single Sign-On.

searchFilter

The String to be combined with the default LDAP user search string to form the value. For example, vpnallowed=true with ldaploginame "samaccount" and the user-supplied username "bob" would yield the LDAP search string "(&(vpnallowed=true)(samaccount=bob)".

groupAttrName

The Attribute name for group extraction from the LDAP server.

subAttributeName

Subattribute name used for group extraction from the LDAP server.

groupAuthName

To associate AAA users with an AAA group, use the command

"bind AAA group ... -username ...".

You can bind different policies to each AAA group. Use the command

"bind AAA group ... -policy ..."

passwdChange

Accept password change requests.

nestedGroupExtraction

Queries the external LDAP server to determine whether the specified group belongs to another group.

maxNestingLevel

Number of levels up to which the system can query nested LDAP groups.

groupNameIdentifier

LDAP-group attribute that uniquely identifies the group. No two groups on one LDAP server can have the same group name identifier.

groupSearchAttribute

LDAP-group attribute that designates the parent group of the specified group. Use this attribute to search for a group?s parent group.

groupSearchSubAttribute

LDAP-group subattribute that designates the parent group of the specified group. Use this attribute to search for a group?s parent group.

groupSearchFilter

Search-expression that can be specified for sending group-search requests to the LDAP server.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Example

```
> show aaa ldapparams Configured LDAP parameters   Server IP: 127.0.0.1   Port: 389  Timeout: 1     BindDn: cn=Manager,dc=florazel,dc=com login: uid     Bas
```

aaa parameter

Sep 22, 2015

The following operations can be performed on "aaa parameter":

set | unset | show

set aaa parameter

Sets the global AAA configuration. Any configuration settings made at this level overrides configuration settings for the authentication server.

Synopsys

set aaa parameter [-enableStaticPageCaching (YES | NO)] [-enableEnhancedAuthFeedback (YES | NO)] [-defaultAuthType <defaultAuthType>] [-maxAAAUsers <positive_integer>] [-maxLoginAttempts <positive_integer> [-failedLoginTimeout <positive_integer>]] [-aaadnatIp <ip_addr|*>]

Arguments

enableStaticPageCaching

The default state of VPN Static Page caching. If nothing is specified, the default value is set to ON.

Possible values: YES, NO

Default value: STATIC_PAGE_CACHING_ENABLED

enableEnhancedAuthFeedback

Enhanced auth feedback provides more information to the end user about the reason for an authentication failure. The default value is set to ON.

Possible values: YES, NO

Default value: ENHANCED_AUTH_FEEDBACK_DISABLED

defaultAuthType

The default authentication server type.

Possible values: LOCAL, LDAP, RADIUS, TACACS, CERT

Default value: LOCAL_AUTH

maxAAAUsers

Maximum number of concurrent users allowed to log on to VPN simultaneously.

Minimum value: 1

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

aaadnatIp

Source IP address to use for traffic that is sent to the authentication server.

Example

set aaa parameter -defaultAuthType RADIUS -maxAAAUSers 100

---

unset aaa parameter

Resets the global AAA parameter settings on the NetScaler appliance. Attributes for which a default value is available revert to their default values. See the set aaa parameter command for descriptions of the parameters..Refer to the set aaa parameter command for meanings of the arguments.

Synopsys

unset aaa parameter [-enableStaticPageCaching] [-enableEnhancedAuthFeedback] [-defaultAuthType] [-maxAAAUsers] [-aaadnatIp] [-maxLoginAttempts]

---

show aaa parameter

Displays the current AAA global configuration.

Synopsys

show aaa parameter

Arguments

format

level

Outputs

enableStaticPageCaching

Indicates if static page caching is enabled or not.

enableEnhancedAuthFeedback

Indicates whether enhanced auth feedback is enabled or not.

defaultAuthType

The default authentication server type.

maxAAAUsers

The maximum number of concurrent users allowed to log into the system at any time.

aaadnatIp

The natIp to be used for the AAA traffic

maxLoginAttempts

Maximum Number of login Attempts

failedLoginTimeout

Failed Login timeout

Example

```
> show aaa parameter Configured AAA parameters        DefaultAuthType: LDAP   MaxAAAUsers: 5  Done >
```

aaa preauthenticationaction

Sep 22, 2015

The following operations can be performed on "aaa preauthenticationaction":

add | rm | set | unset | show

add aaa preauthenticationaction

Adds an action (profile) for endpoint analysis (EPA) clients before authentication.

Synopsys

add aaa preauthenticationaction <name> [<preauthenticationaction>] [-killProcess <string>] [-deletefiles <string>]

Arguments

name

Name for the preauthentication action. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after preauthentication action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my aaa action? or ?my aaa action).

preauthenticationaction

Allow or deny logon after endpoint analysis (EPA) results.

Possible values: ALLOW, DENY

killProcess

String specifying the name of a process to be terminated by the endpoint analysis (EPA) tool.

deletefiles

String specifying the path(s) and name(s) of the files to be deleted by the endpoint analysis (EPA) tool.

---

rm aaa preauthenticationaction

Removes a preauthentication action. NOTE: A preauthentication action cannot be removed if it is bound to a policy.

Synopsys

rm aaa preauthenticationaction <name>

Arguments

name

Name of the preauthentication action to remove.

---

set aaa preauthenticationaction

Modifies an existing preauthentication action (profile).

Synopsys

set aaa preauthenticationaction <name> [<preauthenticationaction>] [-killProcess <string>] [-deletefiles <string>]

Arguments

name

Name of the preauthentication action to modify.

preauthenticationaction

Allow or deny logon after endpoint analysis (EPA) results.

Possible values: ALLOW, DENY

killProcess

String specifying the name of a process to be terminated by the endpoint analysis (EPA) tool.

deletefiles

String specifying the path(s) and name(s) of the files to be deleted by the endpoint analysis (EPA) tool.

---

unset aaa preauthenticationaction

Use this command to remove aaa preauthenticationaction settings.Refer to the set aaa preauthenticationaction command for meanings of the arguments.

Synopsys

unset aaa preauthenticationaction <name> [-killProcess] [-deletefiles]

---

show aaa preauthenticationaction

Displays details of the specified preauthentication action.

Synopsys

show aaa preauthenticationaction [<name>]

Arguments

name

Name of the preauthentication action.

summary

fullValues

format

level

Outputs

preauthenticationaction

Allow or deny logon after endpoint analysis (EPA) results.

killProcess

String specifying the name of a process to be terminated by the endpoint analysis (EPA) tool.

deletefiles

String specifying the path(s) and name(s) of the files to be deleted by the endpoint analysis (EPA) tool.

stateflag

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

aaa preauthenticationparameter

Sep 22, 2015

The following operations can be performed on "aaa preauthenticationparameter":

set | unset | show

set aaa preauthenticationparameter

Configures the default end point analysis (EPA) parameters that are applied before authentication.

Synopsys

set aaa preauthenticationparameter [-preauthenticationaction (ALLOW | DENY)] [-rule <expression>] [-killProcess <string>] [-deletefiles <string>]

Arguments

preauthenticationaction

Deny or allow login on the basis of end point analysis results.

Possible values: ALLOW, DENY

rule

Name of the NetScaler named rule, or a default syntax expression, to be evaluated by the EPA tool.

killProcess

String specifying the name of a process to be terminated by the EPA tool.

deletefiles

String specifying the path(s) to and name(s) of the files to be deleted by the EPA tool, as a string of between 1 and 1023 characters.

---

unset aaa preauthenticationparameter

Resets the default end point analysis(EPA) configuration settings on the NetScaler appliance. Attributes for which a default value is available revert to their default values. See the set aaa preauthenticationparameter command for descriptions of the parameters..Refer to the set aaa preauthenticationparameter command for meanings of the arguments.

Synopsys

unset aaa preauthenticationparameter [-rule] [-preauthenticationaction] [-killProcess] [-deletefiles]

---

show aaa preauthenticationparameter

Displays the current preauthentication configuration.

Synopsys

show aaa preauthenticationparameter

Arguments

format

level

Outputs

preauthenticationaction

Deny or allow login after End point analysis results.

rule

Name of the NetScaler named rule, or a default syntax expression, to be evaluated by the EPA tool.

killProcess

Processes to be killed by EPA tool.

deletefiles

Files to be deleted by EPA tool.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

aaa preauthenticationpolicy

Sep 22, 2015

The following operations can be performed on "aaa preauthenticationpolicy":

add | rm | set | show

add aaa preauthenticationpolicy

Adds a preauthentication policy. The policy defines expressions to be evaluated by the endpoint analysis (EPA) tool.

Synopsys

add aaa preauthenticationpolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the preauthentication policy. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the preauthentication policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my policy? or ?my policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, defining connections that match the policy.

reqAction

Name of the action that the policy is to invoke when a connection matches the policy.

---

rm aaa preauthenticationpolicy

Removes the specified preauthentication policy.

Synopsys

rm aaa preauthenticationpolicy <name>

Arguments

name

Name of the preauthentication policy to remove.

---

set aaa preauthenticationpolicy

Modifies the Request Action of a preauthentication policy.

Synopsys

set aaa preauthenticationpolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the preauthentication policy to modifiy.

rule

The new rule to be associated with the policy.

reqAction

Name of the action that the policy is to invoke when a connection matches the policy.

---

show aaa preauthenticationpolicy

Displays the properties of either the specified preauthentication policy or (if none is specified) a list of all configured preauthentication policies.

Synopsys

show aaa preauthenticationpolicy [<name>]

Arguments

name

Name of the preauthentication policy whose properties you want to view.

summary

fullValues

format

level

Outputs

rule

The new rule associated with the policy.

reqAction

The Pre-authentication action associated with the policy.

hits

No of hits.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

aaa radiusParams

Sep 22, 2015

The following operations can be performed on "aaa radiusParams":

set | unset | show

set aaa radiusParams

Modifies the global configuration settings for the RADIUS server. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

set aaa radiusParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip (ENABLED | DISABLED)] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting (ON | OFF)] [-pwdVendorID <positive_integer>] [-pwdAttributeType <positive_integer>] [-defaultAuthenticationGroup <string>] [-callingstationid (ENABLED | DISABLED)]

Arguments

serverIP

IP address of your RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

Default value: 1812

Minimum value: 1

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

radKey

The key shared between the RADIUS server and clients.

Required for allowing the NetScaler appliance to communicate with the RADIUS server.

radNASip

Send the NetScaler IP (NSIP) address to the RADIUS server as the Network Access Server IP (NASIP) part of the Radius protocol.

Possible values: ENABLED, DISABLED

radNASid

Send the Network Access Server ID (NASID) for your NetScaler appliance to the RADIUS server as the nasid part of the Radius protocol.

radVendorID

Vendor ID for RADIUS group extraction.

Minimum value: 1

radAttributeType

Attribute type for RADIUS group extraction.

Minimum value: 1

radGroupsPrefix

Prefix string that precedes group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

Group separator string that delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Enable password encoding in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

Possible values: pap, chap, mschapv1, mschapv2

Default value: AAA_PAP

ipVendorID

Vendor ID attribute in the RADIUS response.

If the attribute is not vendor-encoded, it is set to 0.

ipAttributeType

IP attribute type in the RADIUS response.

Minimum value: 1

accounting

Configure the RADIUS server state to accept or refuse accounting messages.

Possible values: ON, OFF

pwdVendorID

Vendor ID of the password in the RADIUS response. Used to extract the user password.

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

To configure the default RADIUS parameters: set aaa radiusparams -serverip 192.30.1.2 -radkey sslvpn

---

unset aaa radiusParams

Use this command to remove aaa radiusParams settings.Refer to the set aaa radiusParams command for meanings of the arguments.

Synopsys

unset aaa radiusParams [-serverIP] [-serverPort] [-authTimeout] [-radNASip] [-radNASid] [-radVendorID] [-radAttributeType] [-radGroupsPrefix] [-radGroupSeparator] [-passEncoding] [-ipVendorID] [-ipAttributeType] [-accounting] [-pwdVendorID] [-pwdAttributeType] [-defaultAuthenticationGroup] [-callingstationid]

---

show aaa radiusParams

Displays the current RADIUS configuration on the NetScaler appliance.

Synopsys

show aaa radiusParams

Arguments

format

level

Outputs

serverIP

IP address of your RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

radKey

The key shared between the client and the server.

groupAuthName

To associate AAA users with an AAA group, use the command

"bind AAA group ... -username ...".

You can bind different policies to each AAA group. Use the command

"bind AAA group ... -policy ..."

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the RADIUS server.

radNASip

The option to send the NetScaler's IP address (NSIP) as the "nasip" (Network Access Server IP) part of the Radius protocol to the server.

radNASid

The nasid (Network Access Server ID). If configured, this string will be sent to the RADIUS server as the "nasid" as part of the Radius protocol.

IPAddress

IP Address.

radVendorID

Vendor ID for RADIUS group extraction.

radAttributeType

Attribute type for RADIUS group extraction.

radGroupsPrefix

Prefix string that precedes group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

Group separator string that delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Enable password encoding in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

ipVendorID

Vendor ID attribute in the RADIUS response.

If the attribute is not vendor-encoded, it is set to 0.

ipAttributeType

IP attribute type in the RADIUS response.

accounting

The state of the Radius server that will receive accounting messages.

pwdVendorID

Vendor ID of the password in the RADIUS response. Used to extract the user password.

pwdAttributeType

Attribute type of the Vendor ID in the RADIUS response.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Example

> show aaa radiusparams Configured RADIUS parameters Server IP: 127.0.0.2 Port: 1812 key: secret Timeout: 10 Done >

aaa session

Sep 22, 2015

The following operations can be performed on "aaa session":

show | kill

show aaa session

Displays all AAA-TM/VPN connections that are bound to the specified user, group, IP address, or IP range.

Synopsys

show aaa session [-userName <string>] [-groupName <string>] [-intranetIP <ip_addr|*> [<netmask>]]

Arguments

userName

Name of the AAA user.

groupName

Name of the AAA group.

intranetIP

IP address or the first address in the intranet IP range.

Outputs

publicIP

Client's public IP address

publicPort

Client's public port

IPAddress

NetScaler's IP address

port

NetScaler's port

privateIP

Client's private/mapped IP address

privatePort

Client's private/mapped port

destIP

Destination IP address

destPort

Destination port

intranetIP

Specifies the Intranet IP

peId

Core id of the session owner

stateflag

devno

count

Example

> show aaa connection ClintIp (ClientPort) -> ServerIp(ServerPort) ---------- ---------- User Name: Joe 10.102.0.39

---

kill aaa session

Terminates the specified AAA-TM/VPN session.

Synopsys

kill aaa session [-userName <string>] [-groupName <string>] [-intranetIP <ip_addr|*> [<netmask>]] [-all]

Arguments

userName

Terminate AAA-TM/VPN sessions that belong to the specified user.

groupName

Terminate AAA-TM/VPN sessions that belong to any user that is a member of the specified group.

intranetIP

Terminate AAA-TM/VPN sessions that are associated with the specified intranet IP address or with an address in the range specified by the address and subnet mask.

all

Terminate all active AAA-TM/VPN sessions.

Example

```
kill aaa session -user joe
```

aaa stats

Sep 22, 2015

The following operations can be performed on "aaa stats":

show aaa stats

show aaa stats is an alias for stat aaa

Synopsys

show aaa stats - alias for 'stat aaa'

aaa tacacsParams

Sep 22, 2015

The following operations can be performed on "aaa tacacsParams":

set | unset | show

set aaa tacacsParams

Modifies the global configuration settings for the TACACS+ server. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

set aaa tacacsParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-tacacsSecret } [-authorization (ON | OFF)] [-accounting (ON | OFF)] [-auditFailedCmds (ON | OFF)] [-defaultAuthenticationGroup <string>]

Arguments

serverIP

IP address of your TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

Default value: 49

Minimum value: 1

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the TACACS+ server.

Default value: 3

Minimum value: 1

tacacsSecret

Key shared between the TACACS+ server and clients. Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

Possible values: ON, OFF

accounting

Send accounting messages to the TACACS+ server.

Possible values: ON, OFF

auditFailedCmds

The option for sending accounting messages to the TACACS+ server.

Possible values: ON, OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

To configure a TACACS+ server running at 192.168.1.20 set aaa tacacsparams -serverip 192.168.1.20 -tacacssecret secret

---

unset aaa tacacsParams

Use this command to remove aaa tacacsParams settings.Refer to the set aaa tacacsParams command for meanings of the arguments.

Synopsys

unset aaa tacacsParams [-serverIP] [-serverPort] [-authTimeout] [-tacacsSecret] [-authorization] [-accounting] [-auditFailedCmds] [-defaultAuthenticationGroup]

---

show aaa tacacsParams

Displays the NetScaler appliance?s current AAA TACACS+ configuration.

Synopsys

show aaa tacacsParams

Arguments

format

level

Outputs

serverIP

IP address of your TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the TACACS+ server.

tacacsSecret

The key shared between the client and the server.

authorization

The option for the streaming authorization for TACACS+ server.

accounting

The option to send accounting messages to TACACS+ server.

auditFailedCmds

The option to send accounting messages to TACACS+ server.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Example

```
> sh aaa tacacsparams Configured TACACS parameter        Server IP: 192.168.1.20 Port: 49   Timeout: 1 secs Done
```

aaa user

Sep 22, 2015

The following operations can be performed on "aaa user":

add | rm | set | bind | unbind | show

add aaa user

Adds a local AAA user account and verifies the configuration to ensure that it is correct.

Synopsys

add aaa user <userName> {-password }

Arguments

userName

Name for the user. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the user is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or

single quotation marks (for example, "my aaa user" or "my aaa user").

password

Password with which the user logs on. Required for any user account that does not exist on an external authentication server.

If you are not using an external authentication server, all user accounts must have a password. If you are using an external authentication server, you must provide a password for local user accounts that do not exist on the authentication server.

Example

add aaa user johndoe -password abcd add aaa user johndoe -password The above example adds user johndoe with password abcd for first case, password supplied on pror

---

rm aaa user

Removes a local AAA user account and the associated configuration.

Synopsys

rm aaa user <userName>

Arguments

userName

Name of the AAA user account to remove.

---

set aaa user

Configures the password for an existing local AAA user account. This command prompts you for a new password. NOTE: AAA does not request confirmation of the new password, so you might want to test the new password before sending it to the user.

Synopsys

set aaa user <userName>

Arguments

userName

Name of the local AAA user account.

password

Password with which the user logs on. Required for any user account that does not exist on an external authentication server.

If you are not using an external authentication server, all user accounts must have a password. If you are using an external authentication server, you must provide a password for local user accounts that do not exist on the authentication server.

Example

set aaa user johndoe password abcd The above command sets the password for johndoe to abcd

---

bind aaa user

Binds a policy to the specified user account.

Synopsys

bind aaa user <userName> [-policy <string> [-priority <positive_integer>]] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> [<netmask>]]

Arguments

userName

User account to which to bind the policy.

policy

Name for the policy that you are creating. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or

single quotation marks (for example, "my policy" or "my policy").

intranetApplication

Name of the intranet VPN application to which the policy applies.

urlName

URL of the intranet application to which you are binding the policy.

intranetIP

IP address of the intranet application to which you are binding the policy.

Example

To bind intranetip to the user joe: bind aaa user joe -intranetip 10.102.1.123

---

unbind aaa user

Unbinds a policy from the specified user account.

Synopsys

unbind aaa user <userName> [-policy <string>] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> [<netmask>]]

Arguments

userName

Name of the user account from which to unbind the policy.

policy

Name of the policy to unbind.

intranetApplication

Name of the intranet VPN application from which you are unbinding the policy.

urlName

URL of the intranet application from which you are unbinding the policy.

intranetIP

Intranet IP address of the application from which you are unbinding the policy.

Example

unbind AAA user joe -intranetip 10.102.1.123

---

show aaa user

Displays the current configuration of a AAA user account.

Synopsys

show aaa user [<userName>] [-loggedIn]

Arguments

userName

Name of the user who has the account.

loggedIn

Show whether the user is logged in or not.

summary

fullValues

format

level

Outputs

groupName

The group name

policy

The policy Name.

priority

The priority of the policy.

intranetApplication

Name of the intranet VPN application to which the policy applies.

urlName

The intranet url.

actType

intranetIP

The Intranet IP bound to the user

netmask

The netmask for the Intranet IP

policySubType

stateflag

password

Password with which the user logs on. Required for any user account that does not exist on an external authentication server.

If you are not using an external authentication server, all user accounts must have a password. If you are using an external authentication server, you must provide a password for local user accounts that do not exist on the authentication server.

devno

count

Example

Example > show aaa user joe UserName: joe IntranetIP: 10.102.1.123 Bound to groups: Group

Application Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- application

application

Sep 22, 2015

The following operations can be performed on "application":

import | export | rm

import application

Imports application configuration information from an AppExpert application template file. You can specify a deployment file along with the template file. A template file contains application and variable definitions. A deployment file contains information about the services, service groups, endpoints, and variables that were in the AppExpert application configuration at the time the template file was created. Before you use template and deployment files, make sure that they are present in the /nsconfig/nstemplates/applications/ and /nsconfig/nstemplates/applications/deployment_files directories, respectively. You can transfer the files from your local drive to those directories on the NetScaler appliance by using either FTP or the NetScaler configuration utility. In the configuration utility, you can also import the files and create the application by using a single wizard (AppExpert > Applications > Import > AppExpert Template Wizard).

Synopsys

import application <apptemplateFilename> [-appname <string>] [-deploymentFilename <input_filename>]

Arguments

apptemplateFilename

Name of the AppExpert application template file.

appname

Name to assign to the application on the NetScaler appliance. If you do not provide a name, the appliance assigns the application the name of the template file.

deploymentFilename

Name of the deployment file.

Example

```
import app application sampleapp -apptemplatefilename sampleapp.xml -deploymentfilename deploy.xml
```

---

export application

Exports application configuration information to an AppExpert application template file. A deployment file is created along with the template file. The template file contains application and variable definitions. The deployment file contains information about the services, service groups, endpoints, and variables that are in the AppExpert application configuration. The template and deployment files are exported to the /nsconfig/nstemplates/applications/ and /nsconfig/nstemplates/applications/deployment_files directories, respectively. If you use the configuration utility, you can also export an application to your local hard drive.

Synopsys

export application <appname> [-apptemplateFilename <input_filename>] [-deploymentFilename <input_filename>]

Arguments

appname

Name of the AppExpert application whose configuration you want to export to a template file.

apptemplateFilename

Name with which to save the template file. If you do not specify a name, the template file is saved with the name of the application.

deploymentFilename

Name with which to save the deployment file. If you do not specify a name, a string consisting of an underscore and ?deployment? (_deployment) is automatically appended to the name of the template file to create the name of the deployment file.

---

rm application

Remove application configuration information from a netscaler device. You can specify an application name as input. All the configuration belonging to the specified application will be removed from the device.

Synopsys

rm application <appname>

Arguments

appname

Name of the AppExpert application whose configuration you want to remove from the Netscaler appliance.

AppFlow Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- appflow
- appflow action
- appflow collector
- appflow global
- appflow param
- appflow policy
- appflow policylabel

appflow

Sep 22, 2015

The following operations can be performed on "appflow":

stat appflow

Display AppFlow statistics.

Synopsys

stat appflow [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

AppFlow octets transmitted (aflwOcts)

The total number of AppFlow (IPFIX) octets that the NetScaler has transmitted.

AppFlow flows transmitted (aflwFlws)

The total number of AppFlow (IPFIX) flows that the NetScaler has transmitted.

AppFlow messages transmitted (aflwMsgs)

The total number of AppFlow (IPFIX) messages that the NetScaler has transmitted.

Octets ignored for AppFlow (aflwIgnOct)

The total number of octets that the NetScaler has ignored for AppFlow (IPFIX).

Packets ignored for AppFlow (aflwIgnPkts)

The total number of packets that the NetScaler has ignored for AppFlow (IPFIX).

AppFlow octets not transmitted (aflwNoTxOcts)

The total number of AppFlow (IPFIX) octets that the NetScaler has not transmitted.

AppFlow flows not transmitted (aflwNoTxFlows)

The total number of AppFlow (IPFIX) flows that the NetScaler has not transmitted.

AppFlow packets not transmitted (aflwNoTxPkts)

The total number of AppFlow (IPFIX) packets that the NetScaler has not transmitted.

appflow action

Sep 22, 2015

The following operations can be performed on "appflow action":

add | rm | set | unset | rename | show

add appflow action

Creates an AppFlow action. The action can be associated with an AppFlow policy by using the add appflow policy command.

Synopsys

add appflow action <name> -collectors <string> ... [-comment <string>]

Arguments

name

Name for the action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow action" or 'my appflow action').

collectors

Name(s) of collector(s) to be associated with the AppFlow action.

comment

Any comments about this action. In the CLI, if including spaces between words, enclose the comment in quotation marks. (The quotation marks are not required in the configuration utility.)

Example

```
add appflow action appflow_action_1 -collectors col1 col2
```

---

rm appflow action

Removes a configured AppFlow action. You cannot remove an action that is associated with an AppFlow policy.

Synopsys

rm appflow action <name>

Arguments

name

Name of the action to be removed.

Example

rm appflow action appflow_action_1

---

set appflow action

Modifies the specified parameters of an AppFlow action.

Synopsys

set appflow action <name> [-collectors <string> ...] [-comment <string>]

Arguments

name

Name of the action to be modified.

collectors

Name(s) of collector(s) to be associated with the AppFlow action.

comment

Any comments about this action. In the CLI, if including spaces between words, enclose the comment in quotation marks. (The quotation marks are not required in the configuration utility.)

Example

set appflow action appflow_action_1 -collectors col1 col2 col3

---

unset appflow action

Use this command to remove appflow action settings.Refer to the set appflow action command for meanings of the arguments.

Synopsys

unset appflow action <name> -comment

---

rename appflow action

Renames an AppFlow action.

Synopsys

rename appflow action <name>@ <newName>@

Arguments

name

Existing name of the action.

newName

New name for the AppFlow action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow action" or 'my appflow action').

Example

```
rename appflow action old_name new_name
```

---

show appflow action

Displays information about AppFlow action(s), or about the specified AppFlow action.

Synopsys

```
show appflow action [<name>]
```

Arguments

name

Name of the action about which to display information.

summary

fullValues

format

level

Outputs

stateflag

hits

The number of times the action has been taken.

collectors

Name(s) of collector(s) to be associated with the AppFlow action.

referenceCount

The number of references to the action.

description

Description of the action

comment

Comments associated with the AppFlow action.

devno

count

Example

1. show appflow action 2. show appflow action appflow_action_1

appflow collector

Sep 22, 2015

The following operations can be performed on "appflow collector":

add | rm | rename | show

add appflow collector

Adds a new AppFlow collector. A collector receives the flow records generated by the NetScaler appliance. You can add only four AppFlow collectors to the NetScaler appliance.

Synopsys

add appflow collector <name> -IPAddress <ip_addr> [-port <port>] [-netProfile <string>]

Arguments

name

Name for the collector. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

Only four collectors can be configured.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow collector" or 'my appflow collector').

IPAddress

IPv4 address of the collector.

port

UDP port on which the collector listens.

Default value: 4739

netProfile

Netprofile to associate with the collector. The IP address defined in the profile is used as the source IP address for AppFlow traffic for this collector. If you do not set this parameter, the NetScaler IP (NSIP) address is used as the source IP address.

Example

```
add appflow collector collector1 -IPAddress 192.168.1.40 -port 2055
```

rm appflow collector

Removes an AppFlow collector. You cannot remove a collector if it is associated with an AppFlow action.

Synopsys

rm appflow collector <name>

Arguments

name

Name of the collector to remove.

Example

```
rm appflow collector collector1
```

---

rename appflow collector

Renames an AppFlow collector.

Synopsys

rename appflow collector <name>@ <newName>@

Arguments

name

Existing name of the collector.

newName

New name for the collector. Must begin with an ASCII alphabetic or underscore (_) character, and must

contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow coll" or 'my appflow coll').

Example

```
rename appflow collector old_name new_name
```

---

show appflow collector

Displays information about all configured AppFlow collectors, or about the specified collector.

Synopsys

show appflow collector [<name>]

Arguments

name

Name of the collector about which to display information.

summary

fullValues

format

level

Outputs

IPAddress

IPv4 address of the collector.

port

UDP port on which the collector listens.

netProfile

Netprofile to associate with the collector. The IP address defined in the profile is used as the source IP address for AppFlow traffic for this collector. If you do not set this parameter, the NetScaler IP (NSIP) address is used as the source IP address.

devno

count

stateflag

Example

```
show appflow collector collector1
```

appflow global

Sep 22, 2015

The following operations can be performed on "appflow global":

bind | unbind | show

bind appflow global

Binds the AppFlow policy to one of the two global lists of AppFlow policies. A policy becomes active only after it is bound.

Synopsys

bind appflow global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]

Arguments

policyName

Name of the AppFlow policy to be bound.

Example

i) bind appflow global pol9 9 ii) bind appflow global pol9 9 120 iii) bind appflow global pol9 9 "HTTP.REQ.HEADER(\\\\"qh3\\\\").TYPECAST_NUM_T(DECIMAL)"

---

unbind appflow global

Unbinds entities from an AppFlow global bind point.

Synopsys

unbind appflow global (<policyName> [-type <type>] [-priority <positive_integer>])

Arguments

policyName

Name of the policy to be unbound.

Example

unbind appflow global pol9

---

show appflow global

Displays the AppFlow global bind points and the number of policies bound to each global bind point, or more detailed information about the specified bind point.

Synopsys

show appflow global [-type <type>]

Arguments

type

Global bind point for which to show detailed information about the policies bound to the bind point.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, ICA_REQ_OVERRIDE, ICA_REQ_DEFAULT, ORACLE_REQ_OVERRIDE, ORACLE_REQ_DEFAULT

summary

fullValues

format

level

Outputs

stateflag

policyName

Name of the AppFlow policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to invoke. Specify vserver for a policy label associated with a virtual server, or policylabel for a user-defined policy label.

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

numpol

The number of policies bound to the bindpoint.

flowType

Flow type of the bound AppFlow policy.

flags

devno

count

Example

show appflow global

appflow param

Sep 22, 2015

The following operations can be performed on "appflow param":

set | unset | show

set appflow param

Configures AppFlow parameters.

Synopsys

set appflow param [-templateRefresh <secs>] [-appnameRefresh <secs>] [-flowRecordInterval <secs>] [-udpPmtu <positive_integer>] [-httpUrl (ENABLED | DISABLED)] [-AAAUserName (ENABLED | DISABLED)] [-httpCookie (ENABLED | DISABLED)] [-httpReferer (ENABLED | DISABLED)] [-httpMethod (ENABLED | DISABLED)] [-httpHost (ENABLED | DISABLED)] [-httpUserAgent (ENABLED | DISABLED)] [-clientTrafficOnly (YES | NO)] [-httpContentType (ENABLED | DISABLED)] [-httpAuthorization (ENABLED | DISABLED)] [-httpVia (ENABLED | DISABLED)] [-httpXForwardedFor (ENABLED | DISABLED)] [-httpLocation (ENABLED | DISABLED)] [-httpSetCookie (ENABLED | DISABLED)] [-httpSetCookie2 (ENABLED | DISABLED)] [-connectionChaining (ENABLED | DISABLED)]

Arguments

templateRefresh

Refresh interval, in seconds, at which to export the template data. Because data transmission is in UDP, the templates must be resent at regular intervals.

Default value: 600

Minimum value: 60

Maximum value: 3600

appnameRefresh

Interval, in seconds, at which to send Appnames to the configured collectors. Appname refers to the name of an entity (virtual server, service, or service group) in the NetScaler appliance.

Default value: 600

Minimum value: 60

Maximum value: 3600

flowRecordInterval

Interval, in seconds, at which to send flow records to the configured collectors.

Default value: 60

Minimum value: 60

Maximum value: 3600

udpPmtu

MTU, in bytes, for IPFIX UDP packets.

Default value: 1472

Minimum value: 128

Maximum value: 1472

httpUrl

Include the http URL that the NetScaler appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

AAAUserName

Enable AppFlow AAA Username logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpCookie

Include the cookie that was in the HTTP request the appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpReferer

Include the web page that was last visited by the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpMethod

Include the method that was specified in the HTTP request that the appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpHost

Include the host identified in the HTTP request that the appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpUserAgent

Include the client application through which the HTTP request was received by the NetScaler appliance.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientTrafficOnly

Generate AppFlow records for only the traffic from the client.

Possible values: YES, NO

Default value: NO

httpContentType

Include the HTTP Content-Type header sent from the server to the client to determine the type of the content sent.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpAuthorization

Include the HTTP Authorization header information.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpVia

Include the httpVia header which contains the IP address of proxy server through which the client accessed the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpXForwardedFor

Include the httpXForwardedFor header, which contains the original IP Address of the client using a proxy server to access the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpLocation

Include the HTTP location headers returned from the HTTP responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpSetCookie

Include the Set-cookie header sent from the server to the client in response to a HTTP request.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpSetCookie2

Include the Set-cookie header sent from the server to the client in response to a HTTP request.

Possible values: ENABLED, DISABLED

Default value: DISABLED

connectionChaining

Enable connection chaining so that the client server flows of a connection are linked. Also the connection chain ID is propagated across NetScalers, so that in a multi-hop environment the flows belonging to the same logical connection are linked. This id is also logged as part of appflow record

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set appflow param -templateRefresh 240
```

---

unset appflow param

Use this command to remove appflow param settings.Refer to the set appflow param command for meanings of the arguments.

Synopsys

unset appflow param [-templateRefresh] [-appnameRefresh] [-flowRecordInterval] [-udpPmtu] [-httpUrl] [-AAAUserName] [-httpCookie] [-httpReferer] [-httpMethod] [-httpHost] [-httpUserAgent] [-clientTrafficOnly] [-httpContentType] [-httpAuthorization] [-httpVia] [-httpXForwardedFor] [-httpLocation] [-httpSetCookie] [-httpSetCookie2] [-connectionChaining]

---

show appflow param

Displays AppFlow parameters.

Synopsys

show appflow param

Arguments

summary

fullValues

format

level

Outputs

templateRefresh

Refresh interval, in seconds, at which to export the template data. Because data transmission is in UDP, the templates must be resent at regular intervals.

appnameRefresh

Interval, in seconds, at which to send Appnames to the configured collectors. Appname refers to the name of an entity (virtual server, service, or service group) in the NetScaler appliance.

flowRecordInterval

Interval, in seconds, at which to send flow records to the configured collectors.

udpPmtu

MTU, in bytes, for IPFIX UDP packets.

httpUrl

State of AppFlow HTTP URL logging.

AAAUserName

State of AppFlow AAA User logging.

httpCookie

State of AppFlow HTTP cookie logging.

httpReferer

State of AppFlow HTTP referer logging.

httpMethod

State of AppFlow HTTP method logging.

httpHost

State of AppFlow HTTP host logging.

httpUserAgent

State of AppFlow HTTP user-agent logging.

clientTrafficOnly

Generate AppFlow records for only the traffic from the client.

httpContentType

State of AppFlow HTTP Content-Type header logging

httpAuthorization

State of AppFlow HTTP Authorization header logging

httpVia

State of AppFlow HTTP Via header logging

httpXForwardedFor

State of AppFlow HTTP X-Forwarded-For header logging

httpLocation

State of AppFlow HTTP Location header logging

httpSetCookie

State of AppFlow HTTP Setcookie header logging

httpSetCookie2

State of AppFlow HTTP Setcookie2 header logging

connectionChaining

State of connection-chaining feature

appflow policy

Sep 22, 2015

The following operations can be performed on "appflow policy":

add | rm | set | unset | rename | show

add appflow policy

Adds an Appflow policy. The policy specifies the rule based on which the traffic is evaluated, and the action to be taken if the rule returns "TRUE".

Synopsys

add appflow policy <name> <rule> <action> [-comment <string>]

Arguments

name

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policy" or 'my appflow policy').

rule

Expression or other value against which the traffic is evaluated. Must be a Boolean, default syntax expression. Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to be associated with this policy.

comment

Any comments about this policy.

Example

```
add appflow policy appflow_pol "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh3\\\")" appflow_act
```

---

rm appflow policy

Removes an AppFlow policy. (Cannot remove a policy that is bound to a policy label.)

Synopsys

```
rm appflow policy <name>
```

Arguments

name

Name of the policy to be removed.

Example

```
rm appflow policy appflow_policy_1
```

---

set appflow policy

Modifies the rule and/or action for an existing AppFlow policy. The rule for flow type can be changed only if the associated action is of NEUTRAL flow type.

Synopsys

```
set appflow policy <name> [-rule <expression>] [-action <string>] [-comment <string>]
```

Arguments

name

Name of the policy to modify.

rule

Expression or other value against which the traffic is evaluated. Must be a Boolean, default syntax expression. Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to be associated with this policy.

comment

Any comments about this policy.

Example

```
set appflow policy appflow_policy -rule "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"
```

---

unset appflow policy

Use this command to remove appflow policy settings.Refer to the set appflow policy command for meanings of the arguments.

Synopsys

```
unset appflow policy <name> -comment
```

---

rename appflow policy

Renames an AppFlow policy.

Synopsys

```
rename appflow policy <name>@ <newName>@
```

Arguments

name

Existing name of the policy.

newName

New name for the policy. Must begin with an ASCII alphabetic or underscore (_)character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policy" or 'my appflow policy').

Example

rename appflow policy old_name new_name

---

show appflow policy

Displays information about all configured AppFlow policies, or detailed information about the specified policy.

Synopsys

show appflow policy [<name>]

Arguments

name

Name of the policy about which to display detailed information.

summary

fullValues

format

level

Outputs

stateflag

rule

Expression to be used by AppFlow policy.

action

AppFlow action associated with the policy.

hits

Number of hits.

undefHits

Number of policy UNDEF hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

comment

Any comments about this policy.

bindPolicyType

vserverType

devno

count

Example

```
show appflow policy
```

appflow policylabel

Sep 22, 2015

The following operations can be performed on "appflow policylabel":

add | rm | bind | unbind | rename | show

add appflow policylabel

Creates a user-defined AppFlow policy label. You can bind AppFlow policies to the AppFlow policy label.

Synopsys

add appflow policylabel <labelName> [-policylabeltype (HTTP | OTHERTCP)]

Arguments

labelName

Name of the AppFlow policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policylabel" or 'my appflow policylabel').

policylabeltype

Type of traffic evaluated by the policies bound to the policy label.

Possible values: HTTP, OTHERTCP

Default value: NS_PLTMAP_APPFLOW_REQ

Example

```
add appflow policylabel appflow_pol_label
```

---

rm appflow policylabel

Removes an AppFlow policy label.

Synopsys

rm appflow policylabel <labelName>

Arguments

labelName

Name of the policy label to be removed.

Example

```
rm appflow policylabel appflow_pol_label
```

---

bind appflow policylabel

Binds an AppFlow policy to an AppFlow policy label.

Synopsys

bind appflow policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)]

Arguments

labelName

Name of the policy label to which to bind the policy.

policyName

Name of the policy to bind to the policy label.

Example

```
bind appflow policylabel appflow_pol_label -policyName appflow_pol -priority 1
```

---

unbind appflow policylabel

Unbinds an AppFlow policy from an AppFlow policy label.

Synopsys

unbind appflow policylabel <labelName> <policyName> [-priority <positive_integer>]

Arguments

labelName

Name of the policy label from which to unbind a policy.

policyName

Name of the policy to unbind.

Example

```
unbind appflow policylabel appflow_pol_label appflow_pol
```

---

rename appflow policylabel

Renames an AppFlow policy label.

Synopsys

rename appflow policylabel <labelName>@ <newName>@

Arguments

labelName

Existing name of the policylabel.

newName

New name for the policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policylabel" or 'my appflow policylabel')

Example

```
rename appflow policylabel old_name new_name
```

---

show appflow policylabel

Displays information about all AppFlow policy labels, or detailed information about the specified policy label.

Synopsys

```
show appflow policylabel [<labelName>]
```

Arguments

labelName

Name of the policy label about which to display detailed information.

summary

fullValues

format

level

Outputs

stateflag

policylabeltype

Type of traffic evaluated by the policies bound to the policy label.

numpol

Number of polices bound to the policy label.

hits

Number of times the policy label was invoked.

policyName

Name of the AppFlow policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy

rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to be invoked.

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

flowType

Flowtype of the bound AppFlow policy.

description

Description of the policylabel

flags

devno

count

Example

i) show appflow policylabel appflow_pol_label ii) show appflow policylabel

Application Firewall Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- appfw
- appfw JSONContentType
- appfw XMLContentType
- appfw archive
- appfw confidField
- appfw customSettings
- appfw fieldType
- appfw global
- appfw htmlerrorpage
- appfw learningdata
- appfw learningsettings
- appfw policy
- appfw policylabel
- appfw profile
- appfw settings
- appfw signatures
- appfw stats
- appfw transactionRecords
- appfw wsdl
- appfw xmlerrorpage
- appfw xmlschema

appfw

Sep 22, 2015

The following operations can be performed on "appfw":

stat appfw

Displays application firewall statistics.

Synopsys

stat appfw [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

total violations (totviols)

Total number of security check violations seen by the Application Firewall.

Recent Ave Response Time (ms) (shortAvgRespTime)

Average backend response time in milliseconds over the last 7 seconds

Long Term Ave Response Time (ms) (longAvgRespTime)

Average backend response time in milliseconds since reboot

requests (reqs)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

Request Bytes (reqBytes)

Number of bytes transfered for requests

responses (resps)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

Response Bytes (resBytes)

Number of bytes transfered for responses

aborts

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

redirects (redirect)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

Traps Dropped (trapsDr)

AppFirewall SNMP traps dropped due to time limit.

start URL (startURL)

Number of Start URL security check violations seen by the Application Firewall.

deny URL (denyURL)

Number of Deny URL security check violations seen by the Application Firewall.

referer header (refererHdr)

Number of Referer Header security check violations seen by the Application Firewall.

buffer overflow (bufovfl)

Number of Buffer Overflow security check violations seen by the Application Firewall.

cookie consistency (cookie)

Number of Cookie Consistency security check violations seen by the Application Firewall.

CSRF form tag (csrf_tag)

Number of Cross Site Request Forgery form tag security check violations seen by the Application Firewall.

HTML Cross-site scripting (xss)

Number of HTML Cross-Site Scripting security check violations seen by the Application Firewall.

HTML SQL injection (sql)

Number of HTML SQL Injection security check violations seen by the Application Firewall.

field format (fieldfmt)

Number of Field Format security check violations seen by the Application Firewall.

field consistency (fieldcon)

Number of Field Consistency security check violations seen by the Application Firewall.

credit card (ccard)

Number of Credit Card security check violations seen by the Application Firewall.

safe object (safeobj)

Number of Safe Object security check violations seen by the Application Firewall.

Signature Violations (sigs)

Number of Signature violations seen by the Application Firewall.

XML Format (wfcViolations)

Number of XML Format security check violations seen by the Application Firewall.

XML Denial of Service (XDoS) (xdosViolations)

Number of XML Denial-of-Service security check violations seen by the Application Firewall.

XML Message Validation (msgvalViolations)

Number of XML Message Validation security check violations seen by the Application Firewall.

Web Services Interoperability (wsIViolations)

Number of Web Services Interoperability (WS-I) security check violations seen by the Application Firewall.

XML SQL Injection (xmlSqlViolations)

Number of XML SQL Injection security check violations seen by the Application Firewall.

XML Cross-Site Scripting (xmlXssViolations)

Number of XML Cross-Site Scripting (XSS) security check violations seen by the Application Firewall.

XML Attachment (xmlAttachmentViolations)

Number of XML Attachment security check violations seen by the Application Firewall.

SOAP Fault Violations (soapflt)

Number of requests returning soap:fault from the backend server

XML Generic Violations (genflt)

Number of requests returning XML generic error from the backend server

HTTP Client Errors (4xx Resp) (4xxResps)

Number of requests returning HTTP 4xx from the backend server

HTTP Server Errors (5xx Resp) (5xxResps)

Number of requests returning HTTP 5xx from the backend server

appfw JSONContentType

Sep 22, 2015

The following operations can be performed on "appfw JSONContentType":

add | rm | show

add appfw JSONContentType

Add JSON content type. This will classify a request/response with the specified content type as JSON

Synopsys

add appfw JSONContentType <JSONContenttypevalue> [-isRegex (REGEX | NOTREGEX)]

Arguments

JSONContenttypevalue

Content type to be classified as JSON

isRegex

Is json content type a regular expression?

Possible values: REGEX, NOTREGEX

Default value: NS_NOTREGEX

---

rm appfw JSONContentType

Remove JSON content type.

Synopsys

rm appfw JSONContentType <JSONContenttypevalue>

Arguments

JSONContenttypevalue

Content type to be classified as JSON

---

show appfw JSONContentType

Display all JSON content types.

Synopsys

show appfw JSONContentType [<JSONContenttypevalue>]

Arguments

JSONContenttypevalue

Content type to be classified as JSON

summary

fullValues

format

level

Outputs

isRegex

Is json content type a regular expression?

builtin

Flag to determine if jsoncontenttype is built-in or not

devno

count

stateflag

appfw XMLContentType

Sep 22, 2015

The following operations can be performed on "appfw XMLContentType":

add | rm | show

add appfw XMLContentType

Add XML content type. This will classify a request/response with the specified content type as XML

Synopsys

add appfw XMLContentType <XMLContenttypevalue> [-isRegex (REGEX | NOTREGEX)]

Arguments

XMLContenttypevalue

Content type to be classified as XML

isRegex

Is field name a regular expression?

Possible values: REGEX, NOTREGEX

Default value: NS_NOTREGEX

---

rm appfw XMLContentType

Remove XML content type.

Synopsys

rm appfw XMLContentType <XMLContenttypevalue>

Arguments

XMLContenttypevalue

Content type to be classified as XML

---

show appfw XMLContentType

Display all xml content types.

Synopsys

show appfw XMLContentType [<XMLContenttypevalue>]

Arguments

XMLContenttypevalue

Content type to be classified as XML

summary

fullValues

format

level

Outputs

isRegex

Is field name a regular expression?

builtin

Flag to determine if xmlcontenttype is built-in or not

devno

count

stateflag

appfw archive

Sep 22, 2015

The following operations can be performed on "appfw archive":

show | export | import | rm

show appfw archive

Synopsys

show appfw archive

Outputs

response

Example

show appfw archive

---

export appfw archive

Exports the archive file to the specified location

Synopsys

export appfw archive <name> <target>

Arguments

name

Name of tar archive

target

Path to the file to be exported

---

import appfw archive

Imports the archive file from specified location

Synopsys

import appfw archive <src> <name> [-comment <string>]

Arguments

src

Indicates the source of the tar archive file as a URL

of the form

<protocol>://<host>[:<port>][/<path>]

<protocol> is http or https.

<host> is the DNS name or IP address of the http or https server.

<port> is the port number of the server. If omitted, the

default port for http or https will be used.

<path> is the path of the file on the server.

Import will fail if an https server requires client

certificate authentication.

name

Indicates name of archive

comment

Comments associated with this archive.

---

rm appfw archive

Removes the archive created by archive command.

Synopsys

rm appfw archive <name>

Arguments

name

Indicates name of the archive to be removed.

Example

rm appfw archive <name>

---

appfw confidField

Sep 22, 2015

The following operations can be performed on "appfw confidField":

add | rm | set | unset | show

add appfw confidField

Defines the specified web form field as confidential. Form fields designated as confidential have the information that is provided in those fields x'd out in the audit logs.

Synopsys

add appfw confidField <fieldName> <url> [-isRegex (REGEX | NOTREGEX)] [-comment <string>] [-state (ENABLED | DISABLED)]

Arguments

fieldName

Name of the form field to designate as confidential.

url

URL of the web page that contains the web form.

isRegex

Method of specifying the form field name. Available settings function as follows:

* REGEX. Form field is a regular expression.

* NOTREGEX. Form field is a literal string.

Possible values: REGEX, NOTREGEX

Default value: NS_NOTREGEX

comment

Any comments to preserve information about the form field designation.

state

Enable or disable the confidential field designation.

Possible values: ENABLED, DISABLED

Default value: ENABLED

---

rm appfw confidField

Removes a confidential field designation.

Synopsys

rm appfw confidField <fieldName> <url>

Arguments

fieldName

Name of the web form field.

url

URL of the web page that contains the web form in which the field appears.

---

set appfw confidField

Modifies the specified parameters of a confidential field setting. Form fields designated as confidential have the information that is provided in those fields x'd out in the audit logs.

Synopsys

set appfw confidField <fieldName> <url> [-comment <string>] [-state (ENABLED | DISABLED)]

Arguments

fieldName

Name of the field to modify.

url

URL of the web page that contains the web form.

comment

Any comments to preserve information about the form field designation.

state

Enable or disable the confidential field designation.

Possible values: ENABLED, DISABLED

Default value: ENABLED

---

unset appfw confidField

Use this command to remove appfw confidField settings.Refer to the set appfw confidField command for meanings of the arguments.

Synopsys

unset appfw confidField <fieldName> <url> [-comment] [-state]

---

show appfw confidField

Displays the current settings for the specified application firewall confidential field designation. If no confidential field designation is specified, displays a list of all application firewall confidential field designations on the NetScaler appliance.

Synopsys

show appfw confidField [<fieldName> <url>]

Arguments

fieldName

Name of the web form field.

url

URL of the web page that contains the web form with the form field.

summary

fullValues

format

level

Outputs

isRegex

Method of specifying the form field name. Available settings function as follows:

* REGEX. Form field is a regular expression.

* NOTREGEX. Form field is a literal string.

comment

Any comments to preserve information about the form field designation.

state

Enable or disable the confidential field designation.

devno

count

stateflag

appfw customSettings

Sep 22, 2015

The following operations can be performed on "appfw customSettings":

export | rm | show | import | update

export appfw customSettings

NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

target

---

rm appfw customSettings

Removes the object imported by import customsettings. NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

Indicates name of custom-settings object.

Example

rm customsettings <name>

---

show appfw customSettings

Displays the object imported by import customsettings. NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

Outputs

response

Example

show appfw customsettings

---

import appfw customSettings

Downloads the Application Firewall Custom Settings XML configuration to the NetScaler Box with the given object name NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

src

Indicates the source of the custom settings file as a URL

of the form

<protocol>://<host>[:<port>][/<path>]

<protocol> is http or https.

<host> is the DNS name or IP address of the http or https server.

<port> is the port number of the server. If omitted, the

default port for http or https will be used.

<path> is the path of the file on the server.

Import will fail if an https server requires client

certificate authentication.

name

Indicates name of custom-settings object.

comment

Comments.

overwrite

Overwrites the existing file

xslt

XSLT file URL.

merge

Merges the existing Signature with new signature rules

sha1

File path for sha1 file to validate signature file

Example

```
import customsettings http://www.example.com/ns/customsettings.xml my-settings
```

---

update appfw customSettings

Updates the Application Firewall Custom Settings XML configuration to the NetScaler Box with the given object name NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

Indicates name of the custom-settings object to update.

mergeDefault

Merges signature file with default signature file.

Example

```
update customsettings my-settings
```

---

appfw fieldType

Sep 22, 2015

The following operations can be performed on "appfw fieldType":

add | rm | set | show

add appfw fieldType

Adds a field type to the list of field types used by the field format security check. A field type is a regular expression defining the type of data that can appear in a web form field. The Learning engine also uses the field types list to generate appropriate field type assignments for the field formats check.

Synopsys

add appfw fieldType <name> <regex> <priority> [-comment <string>]

Arguments

name

Name for the field type.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at \\(\\@\\), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the field type is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my field type" or 'my field type'\\).

regex

PCRE - format regular expression defining the characters and length allowed for this field type.

priority

Positive integer specifying the priority of the field type. A lower number specified a higher priority. Field types are checked in the order of their priority numbers.

Maximum value: 64000

comment

Comment describing the type of field that this field type is intended to match.

---

rm appfw fieldType

Removes an application firewall field type.

Synopsys

rm appfw fieldType <name>

Arguments

name

Name of the field type.

---

set appfw fieldType

Modifies the properties of the specified application firewall field type.

Synopsys

set appfw fieldType <name> <regex> <priority> [-comment <string>]

Arguments

name

Name for the field type.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at \\(\\@\\), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the field type is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my field type" or 'my field type'\\).

regex

PCRE - format regular expression defining the characters and length allowed for this field type.

---

show appfw fieldType

Displays the regular expression that defines the specified field type and its priority. If no field type is specified, displays all form field types currently configured on the NetScaler appliance.

Synopsys

show appfw fieldType [<name>]

Arguments

name

Name of the field type.

summary

fullValues

format

level

Outputs

regex

PCRE - format regular expression defining the characters and length allowed for this field type.

priority

Positive integer specifying the priority of the field type. A lower number specified a higher priority. Field types are checked in the order of their priority numbers.

comment

Comment describing the type of field that this field type is intended to match.

builtin

Flag to determine if fieldtype is built-in or not

devno

count

stateflag

appfw global

Sep 22, 2015

The following operations can be performed on "appfw global":

bind | unbind | show

bind appfw global

Activates an application firewall policy.

Synopsys

bind appfw global <policyName> <priority> [-state (ENABLED | DISABLED)] [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]

Arguments

policyName

Name of the policy.

---

unbind appfw global

Deactivates the specified application firewall policy. See the bind appfw policy command for descriptions of the parameters.

Synopsys

unbind appfw global <policyName> [-type <type>] [-priority <positive_integer>]

Arguments

policyName

Application Firewall policy name.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

---

show appfw global

Displays a list of application firewall policies that are bound to the specified bind point. If no bind point is specified, displays a list of all application firewall policies

Synopsys

show appfw global [-type <type>]

Arguments

type

Bind point to which to policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, NONE

summary

fullValues

format

level

Outputs

policyName

Name of the policy.

priority

The priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

state

Enable or disable the binding to activate or deactivate the policy.

bindPolicyType

The type of the policy.

policySubType

stateflag

stateflag

labelType

Type of policy label invocation.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

flowType

flowtype of the bound application firewall policy.

numpol

The number of policies bound to the bindpoint.

flags

policyType

flag

devno

count

appfw htmlerrorpage

Sep 22, 2015

The following operations can be performed on "appfw htmlerrorpage":

rm | show | import | update

rm appfw htmlerrorpage

Removes the specified XML error object.

Synopsys

rm appfw htmlerrorpage <name>

Arguments

name

Name of the XML error object to remove.

Example

rm htmlerrorpage <name>

---

show appfw htmlerrorpage

Displays the specified HTML error object. If no HTML error object is specified, lists all HTML error objects on the NetScaler appliance.

Synopsys

show appfw htmlerrorpage [<name>]

Arguments

name

Name of the HTML error object.

Outputs

response

Example

show appfw htmlerrorpage

---

import appfw htmlerrorpage

Imports the specified HTML error page to the NetScaler appliance and assigns it the specified name.

Synopsys

import appfw htmlerrorpage <src> <name> [-comment <string>] [-overwrite]

Arguments

src

URL (protocol, host, path, and name) for the location at which to store the imported HTML error object.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the HTML error object on the NetScaler appliance.

comment

Any comments to preserve information about the HTML error object.

overwrite

Overwrite any existing HTML error object of the same name.

Example

```
import htmlerrorpage http://www.example.com/errorpage.html my-html-error-page
```

---

update appfw htmlerrorpage

Updates the specified HTML error page object from the source.

Synopsys

update appfw htmlerrorpage <name>

Arguments

name

Name of the HTML error page object to update.

Example

```
update htmlerrorpage my-html-error-page
```

---

appfw learningdata

Sep 22, 2015

The following operations can be performed on "appfw learningdata":

rm | show | reset | export

rm appfw learningdata

Removes unreviewed application firewall learning data for the specified application firewall profile.

Synopsys

rm appfw learningdata <profileName> (-startURL <expression> | -cookieConsistency <string> | (-fieldConsistency <string> <formActionURL>) | (-crossSiteScripting <string> <formActionURL>) | (-SQLInjection <string> <formActionURL>) | (-fieldFormat <string> <formActionURL>) | (-CSRFTag <expression> <CSRFFormOriginURL>) | -XMLDoSCheck <expression> | -XMLWSICheck <expression> | -XMLAttachmentCheck <expression>) [-TotalXMLRequests]

Arguments

profileName

Name of the profile.

startURL

Start URL configuration.

cookieConsistency

Cookie Name.

fieldConsistency

Form field name.

crossSiteScripting

Cross-site scripting.

SQLInjection

Form field name.

fieldFormat

Field format name.

CSRFTag

CSRF Form Action URL

XMLDoSCheck

XML Denial of Service check, one of

MaxAttributes

MaxAttributeNameLength

MaxAttributeValueLength

MaxElementNameLength

MaxFileSize

MinFileSize

MaxCDATALength

MaxElements

MaxElementDepth

MaxElementChildren

NumDTDs

NumProcessingInstructions

NumExternalEntities

MaxEntityExpansions

MaxEntityExpansionDepth

MaxNamespaces

MaxNamespaceUriLength

MaxSOAPArraySize

MaxSOAPArrayRank

XMLWSICheck

Web Services Interoperability Rule ID.

XMLAttachmentCheck

XML Attachment Content-Type.

TotalXMLRequests

Total XML requests.

---

show appfw learningdata

Displays the unreviewed application firewall learning data for the specified profile and security check.

Synopsys

show appfw learningdata <profileName> <securityCheck>

Arguments

profileName

Name of the profile.

securityCheck

Name of the security check.

Possible values: startURL, cookieConsistency, fieldConsistency, crossSiteScripting, SQLInjection, fieldFormat, CSRFtag, XMLDoSCheck, XMLWSICheck, XMLAttachmentCheck, TotalXMLRequests

summary

fullValues

Outputs

data

Learned data.

devno

count

stateflag

---

reset appfw learningdata

Remove all databases. Make transaction count zero

Synopsys

reset appfw learningdata

---

export appfw learningdata

Export appfw learnt data in csv format to the location /var/learnt_data/

Synopsys

export appfw learningdata <profileName> <securityCheck> [-target <string>]

Arguments

profileName

Name of the profile.

securityCheck

Name of the security check.

Possible values: startURL, cookieConsistency, fieldConsistency, crossSiteScripting, SQLInjection, fieldFormat, CSRFtag, XMLDoSCheck, XMLWSICheck, XMLAttachmentCheck, TotalXMLRequests

target

Target filename for data to be exported.

appfw learningsettings

Sep 22, 2015

The following operations can be performed on "appfw learningsettings":

set | unset | show

set appfw learningsettings

Configures the application firewall learning settings for the specified profile.

Synopsys

set appfw learningsettings <profileName> [-startURLMinThreshold <positive_integer>] [-startURLPercentThreshold <positive_integer>] [-cookieConsistencyMinThreshold <positive_integer>] [-cookieConsistencyPercentThreshold <positive_integer>] [-CSRFtagMinThreshold <positive_integer>] [-CSRFtagPercentThreshold <positive_integer>] [-fieldConsistencyMinThreshold <positive_integer>] [-fieldConsistencyPercentThreshold <positive_integer>] [-crossSiteScriptingMinThreshold <positive_integer>] [-crossSiteScriptingPercentThreshold <positive_integer>] [-SQLInjectionMinThreshold <positive_integer>] [-SQLInjectionPercentThreshold <positive_integer>] [-fieldFormatMinThreshold <positive_integer>] [-fieldFormatPercentThreshold <positive_integer>] [-XMLWSIMinThreshold <positive_integer>] [-XMLWSIPercentThreshold <positive_integer>] [-XMLAttachmentMinThreshold <positive_integer>] [-XMLAttachmentPercentThreshold <positive_integer>]

Arguments

profileName

Name of the profile.

startURLMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn start URLs.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

startURLPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular start URL pattern for the learning engine to learn that start URL.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

cookieConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cookies.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

cookieConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cookie pattern for the learning engine to learn that cookie.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

CSRFtagMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cross-site request forgery (CSRF) tags.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

CSRFtagPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular CSRF tag for the learning engine to learn that CSRF tag.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

fieldConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field consistency information.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

fieldConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular field consistency pattern for the learning engine to learn that field consistency pattern.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

crossSiteScriptingMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML cross-site scripting patterns.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

crossSiteScriptingPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cross-site scripting pattern for the learning engine to learn that cross-site scripting pattern.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

SQLInjectionMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML SQL injection patterns.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

SQLInjectionPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular HTML SQL injection pattern for the learning engine to learn that HTML SQL injection pattern.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

fieldFormatMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field formats.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

fieldFormatPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular web form field pattern for the learning engine to recommend a field format for that form field.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

XMLWSIMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn web services interoperability (WSI) information.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

XMLWSIPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular pattern for the learning engine to learn a web services interoperability (WSI) pattern.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

XMLAttachmentMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn XML attachment patterns.

Default value: AS_LEARNINGSETTINGS_DEFAULT_MINTHRESHOLD

Minimum value: 1

XMLAttachmentPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular XML attachment pattern for the learning engine to learn that XML attachment pattern.

Default value: AS_LEARNINGSETTINGS_DEFAULT_PERCENTTHRESHOLD

Maximum value: 100

---

unset appfw learningsettings

Use this command to remove appfw learningsettings settings.Refer to the set appfw learningsettings command for meanings of the arguments.

Synopsys

unset appfw learningsettings <profileName> [-startURLMinThreshold] [-startURLPercentThreshold] [-cookieConsistencyMinThreshold] [-cookieConsistencyPercentThreshold] [-CSRFtagMinThreshold] [-CSRFtagPercentThreshold] [-fieldConsistencyMinThreshold] [-fieldConsistencyPercentThreshold] [-crossSiteScriptingMinThreshold] [-crossSiteScriptingPercentThreshold] [-SQLInjectionMinThreshold] [-SQLInjectionPercentThreshold] [-fieldFormatMinThreshold] [-fieldFormatPercentThreshold] [-XMLWSIMinThreshold] [-XMLWSIPercentThreshold] [-XMLAttachmentMinThreshold] [-XMLAttachmentPercentThreshold]

---

show appfw learningsettings

Displays the current application firewall learning settings for the specified profile. If no profile is specified, displays the current application firewall settings for all profiles on the NetScaler appliance.

Synopsys

show appfw learningsettings [<profileName>]

Arguments

profileName

Name of the profile.

summary

fullValues

format

level

Outputs

startURLMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn start URLs.

startURLPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular start URL pattern for the learning engine to learn that start URL.

cookieConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cookies.

cookieConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cookie pattern for the learning engine to learn that cookie.

CSRFtagMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cross-site request forgery (CSRF) tags.

CSRFtagPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular CSRF tag for the learning engine to learn that CSRF tag.

fieldConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field consistency information.

fieldConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular field consistency pattern for the learning engine to learn that field consistency pattern.

crossSiteScriptingMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML cross-site scripting patterns.

crossSiteScriptingPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cross-site scripting pattern for the learning engine to learn that cross-site scripting pattern.

SQLInjectionMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML SQL injection patterns.

SQLInjectionPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular HTML SQL injection pattern for the learning engine to learn that HTML SQL injection pattern.

fieldFormatMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field formats.

fieldFormatPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular web form field pattern for the learning engine to recommend a field format for that form field.

XMLWSIMinThreshold

Minimum threshold to learn XML Web Services Interoperability.

XMLWSIPercentThreshold

Minimum threshold (in percent) to learn XML Web Services Interoperability.

XMLAttachmentMinThreshold

Minimum threshold to learn XML Attachments.

XMLAttachmentPercentThreshold

Minimum threshold (in percent) to learn XML Attachments.

devno

count

stateflag

appfw policy

Sep 22, 2015

The following operations can be performed on "appfw policy":

add | rm | set | unset | show | stat | rename

add appfw policy

Creates an application firewall policy.

Synopsys

add appfw policy <name> <rule> <profileName> [-comment <string>] [-logAction <string>]

Arguments

name

Name for the policy.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore characters. Can be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my policy" or 'my policy'\\).

rule

Name of the NetScaler named rule, or a NetScaler default syntax expression, that the policy uses to determine whether to filter the connection through the application firewall with the designated profile.

profileName

Name of the application firewall profile to use if the policy matches.

comment

Any comments to preserve information about the policy for later reference.

logAction

Where to log information for connections that match this policy.

---

rm appfw policy

Removes an application firewall policy.

Synopsys

rm appfw policy <name>

Arguments

name

Name of the policy to remove.

---

set appfw policy

Modifies the specified parameters of an application firewall policy.

Synopsys

set appfw policy <name> [-rule <expression>] [-profileName <string>] [-comment <string>] [-logAction <string>]

Arguments

name

Name of the policy to modify.

rule

Name of the NetScaler named rule, or a NetScaler default syntax expression, that the policy uses to determine whether to filter the connection through the application firewall with the designated profile.

profileName

Name of the application firewall profile to use if the policy matches.

comment

Any comments to preserve information about the policy for later reference.

logAction

Where to log information for connections that match this policy.

Example

```
set transform policy pol9 -rule "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"
```

---

unset appfw policy

Removes the settings of an existing application firewall policy. Attributes for which a default value is available revert to their default values. See the set appfw policy command for a description of the parameters..Refer to the set appfw policy command for meanings of the arguments.

Synopsys

unset appfw policy <name> [-comment] [-logAction]

Example

```
unset transform policy pol9 -undefAction
```

show appfw policy

Displays the current settings for the specified application firewall policy. If no policy name is provided, displays a list of all application firewall policies currently configured on the NetScaler appliance.

Synopsys

show appfw policy [<name>]

Arguments

name

Name of the policy.

summary

fullValues

format

level

Outputs

stateflag

rule

Name of the NetScaler named rule, or a NetScaler default syntax expression, that the policy uses to determine whether to filter the connection through the application firewall with the designated profile.

profileName

Name of the application firewall profile to use if the policy matches.

hits

Number of hits.

piHits

Number of hits.

undefHits

Number of Undef hits.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any comments to preserve information about the policy for later reference.

logAction

Where to log information for connections that match this policy.

boundTo

The entity name to which policy is bound

activePolicy

Indicates whether policy is bound or not.

priority

Specifies the priority of the policy.

bindPolicyType

policyType

vserverType

devno

count

---

stat appfw policy

Displays statistics for the specified application firewall policy. If no application firewall policy is specified, displays abbreviated statistics for all application firewall policies.

Synopsys

stat appfw policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the application firewall policy.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat appfw policy
```

---

rename appfw policy

Renames an application firewall policy.

Synopsys

```
rename appfw policy <name>@ <newName>@
```

Arguments

name

Existing name of the application firewall policy.

newName

New name for the policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

Example

```
rename appfw policy oldname newname
```

appfw policylabel

Sep 22, 2015

The following operations can be performed on "appfw policylabel":

add | rm | bind | unbind | show | stat | rename

add appfw policylabel

Creates a user-defined application firewall policy label.

Synopsys

add appfw policylabel <labelName> <policylabeltype>

Arguments

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy label" or 'my policy label').

policylabeltype

Type of transformations allowed by the policies bound to the label. Always http_req for application firewall policy labels.

Possible values: http_req

Example

```
add appfw policylabel appfw_label http_req
```

---

rm appfw policylabel

Removes the specified application firewall policy label.

Synopsys

rm appfw policylabel <labelName>

Arguments

labelName

Name of the application firewall policy label to remove.

Example

```
rm appfw policylabel appfw_label
```

---

bind appfw policylabel

Binds the specified application firewall policy to the specified policy label.

Synopsys

bind appfw policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

Arguments

labelName

Name of the application firewall policy label.

policyName

Name of the application firewall policy to bind to the policy label.

Example

i) bind appfw policylabel trans_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind appfw policylabel trans_http_url pol_2 2

---

unbind appfw policylabel

Unbinds the specified application firewall policy from the specified policy label. See the bind appfw policylabel command for descriptions of the parameters.

Synopsys

unbind appfw policylabel <labelName> <policyName> [-priority <positive_integer>]

Arguments

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy label" or 'my policy label').

policyName

Name of the application firewall policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

unbind appfw policylabel appfw_label

---

show appfw policylabel

Displays the current settings for the specified application firewall policy label. If no policy label is specified, displays a list of all application firewall policy labels currently configured on the NetScaler appliance.

Synopsys

show appfw policylabel [<labelName>]

Arguments

labelName

Name of the application firewall policy label.

summary

fullValues

format

level

Outputs

stateflag

policylabeltype

Type of transformations allowed by the policies bound to the label. Always http_req for application firewall policy labels.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the application firewall policy to bind to the policy label.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within a group of policies that are bound to the same bind point or label. Policies are evaluated in the order of their priority numbers.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of policy label to invoke if the current policy evaluates to TRUE and the invoke parameter is set. Available settings function as follows:

* reqvserver. Invoke the unnamed policy label associated with the specified request virtual server.

* policylabel. Invoke the specified user-defined policy label.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

description

Description of the policylabel

flags

policyType

devno

count

Example

i) show appfw policylabel appfw_label ii) show appfw policylabel

---

stat appfw policylabel

Displays statistics for the specified application firewall policy label. If no application firewall policy label is specified, displays abbreviated statistics for all application firewall policy labels.

Synopsys

stat appfw policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

labelName

Name of the application firewall policy label.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

---

rename appfw policylabel

Renames an application firewall policy label.

Synopsys

rename appfw policylabel <labelName>@ <newName>@

Arguments

labelName

Existing name of the application firewall policy label.

newName

The new name of the application firewall policylabel.

Example

```
rename appfw policylabel oldname newname
```

appfw profile

Sep 22, 2015

The following operations can be performed on "appfw profile":

add | rm | set | unset | bind | unbind | show | stat | archive | restore

add appfw profile

Creates an application firewall profile, which specifies how the application firewall should protect a given type of web content. (A profile is equivalent to an action in other NetScaler features.)

Synopsys

add appfw profile <name> [-defaults (basic | advanced)] [-startURLAction <startURLAction> ...] [-contentTypeAction <contentTypeAction> ...] [-startURLClosure (ON | OFF)] [-denyURLAction <denyURLAction> ...] [-RefererHeaderCheck <RefererHeaderCheck>] [-cookieConsistencyAction <cookieConsistencyAction> ...] [-cookieTransforms (ON | OFF)] [-cookieEncryption <cookieEncryption>] [-cookieProxying (none | sessionOnly)] [-addCookieFlags <addCookieFlags>] [-fieldConsistencyAction <fieldConsistencyAction> ...] [-CSRFtagAction <CSRFtagAction> ...] [-crossSiteScriptingAction <crossSiteScriptingAction> ...] [-crossSiteScriptingTransformUnsafeHTML (ON | OFF)] [-crossSiteScriptingCheckCompleteURLs (ON | OFF)] [-SQLInjectionAction <SQLInjectionAction> ...] [-SQLInjectionTransformSpecialChars (ON | OFF)] [-SQLInjectionType <SQLInjectionType>] [-SQLInjectionCheckSQLWildChars (ON | OFF)] [-fieldFormatAction <fieldFormatAction> ...] [-defaultFieldFormatType <string>] [-defaultFieldFormatMinLength <positive_integer>] [-defaultFieldFormatMaxLength <positive_integer>] [-bufferOverflowAction <bufferOverflowAction> ...] [-bufferOverflowMaxURLLength <positive_integer>] [-bufferOverflowMaxHeaderLength <positive_integer>] [-bufferOverflowMaxCookieLength <positive_integer>] [-creditCardAction <creditCardAction> ...] [-creditCard <creditCard> ...] [-creditCardMaxAllowed <positive_integer>] [-creditCardXOut (ON | OFF)] [-requestContentType <string>] [-responseContentType <string>] [-XMLDoSAction <XMLDoSAction> ...] [-XMLFormatAction <XMLFormatAction> ...] [-XMLSQLInjectionAction <XMLSQLInjectionAction> ...] [-XMLSQLInjectionType <XMLSQLInjectionType>] [-XMLSQLInjectionCheckSQLWildChars (ON | OFF)] [-XMLSQLInjectionParseComments <XMLSQLInjectionParseComments>] [-XMLXSSAction <XMLXSSAction> ...] [-XMLWSIAction <XMLWSIAction> ...] [-XMLAttachmentAction <XMLAttachmentAction> ...] [-XMLValidationAction <XMLValidationAction> ...] [-XMLErrorObject <string>] [-signatures <string>] [-XMLSOAPFaultAction <XMLSOAPFaultAction> ...] [-useHTMLErrorObject (ON | OFF)] [-errorURL <expression>] [-HTMLErrorObject <string>] [-logEveryPolicyHit (ON | OFF)] [-stripHtmlComments <stripHtmlComments>] [-stripXmlComments (none | all)] [-exemptClosureURLsFromSecurityChecks (ON | OFF)] [-defaultCharSet <string>] [-postBodyLimit <positive_integer>] [-fileUploadMaxNum <positive_integer>] [-canonicalizeHTMLResponse (ON | OFF)] [-enableFormTagging (ON | OFF)] [-sessionlessFieldConsistency <sessionlessFieldConsistency>] [-sessionlessURLClosure (ON | OFF)] [-semicolonFieldSeparator (ON | OFF)] [-excludeFileUploadFromChecks (ON | OFF)] [-SQLInjectionParseComments <SQLInjectionParseComments>] [-invalidPercentHandling <invalidPercentHandling>] [-type (HTML | XML) ...] [-checkRequestHeaders (ON | OFF)] [-comment <string>]

Arguments

name

Name for the profile. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_)

characters. Cannot be changed after the profile is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

defaults

Default configuration to apply to the profile. Basic defaults are intended for standard content that requires little further configuration, such as static web site content. Advanced defaults are intended for specialized content that requires significant specialized configuration, such as heavily scripted or dynamic content.

CLI users: When adding an application firewall profile, you can set either the defaults or the type, but not both. To set both options, create the profile by using the add appfw profile command, and then use the set appfw profile command to configure the other option.

Possible values: basic, advanced

builtin

Indicates that a profile is a built-in entity.

builtinType

Type of built-in profile. Determines which security checks and settings are used for the profile. (The type specified by the HTML XML setting is also called "Web 2.0.")

CLI users: When adding an application firewall profile, you can set either the defaults or the type, but not both. To set both options, create the profile by using the add appfw profile command, and then use the set appfw profile command to configure the other option.

Possible values: APPFW_NOT_BUILTIN, APPFW_BYPASS, APPFW_BLOCK, APPFW_RESET, APPFW_DROP

startURLAction

One or more Start URL actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -startURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -startURLaction none".

Default value: AS_DEFAULT_DISPOSITION

contentTypeAction

One or more Content-type actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -contentTypeaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -contentTypeaction none".

Default value: AS_DEFAULT_CONTENT_TYPE_DISPOSITION

startURLClosure

Toggle the state of Start URL Closure.

Possible values: ON, OFF

Default value: OFF

denyURLAction

One or more Deny URL actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

NOTE: The Deny URL check takes precedence over the Start URL check. If you enable blocking for the Deny URL check, the application firewall blocks any URL that is explicitly blocked by a Deny URL, even if the same URL would otherwise be allowed by the Start URL check.

CLI users: To enable one or more actions, type "set appfw profile -denyURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -denyURLaction none".

Default value: AS_DEFAULT_DISPOSITION

RefererHeaderCheck

Enable validation of Referer headers.

Referer validation ensures that a web form that a user sends to your web site originally came from your web site, not an outside attacker.

Although this parameter is part of the Start URL check, referer validation protects against cross-site request forgery (CSRF) attacks, not Start URL attacks.

Possible values: OFF, if_present, AlwaysExceptStartURLs, AlwaysExceptFirstRequest

Default value: AS_HEADER_CHECK_OFF

cookieConsistencyAction

One or more Cookie Consistency actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -cookieConsistencyAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -cookieConsistencyAction none".

Default value: AS_NONE

cookieTransforms

Perform the specified type of cookie transformation.

Available settings function as follows:

* Encryption - Encrypt cookies.

* Proxying - Mask contents of server cookies by sending proxy cookie to users.

* Cookie flags - Flag cookies as HTTP only to prevent scripts on user's browser from accessing and possibly modifying them.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cookie transformations. If it is set to OFF, no cookie transformations are performed regardless of any other settings.

Possible values: ON, OFF

Default value: OFF

cookieEncryption

Type of cookie encryption. Available settings function as follows:

* None - Do not encrypt cookies.

* Decrypt Only - Decrypt encrypted cookies, but do not encrypt cookies.

* Encrypt Session Only - Encrypt session cookies, but not permanent cookies.

* Encrypt All - Encrypt all cookies.

Possible values: none, decryptOnly, encryptSessionOnly, encryptAll

Default value: AS_CKI_ENCRYPT_NONE

cookieProxying

Cookie proxy setting. Available settings function as follows:

* None - Do not proxy cookies.

* Session Only - Proxy session cookies by using the NetScaler session ID, but do not proxy permanent cookies.

Possible values: none, sessionOnly

Default value: AS_CKI_PROXY_NONE

addCookieFlags

Add the specified flags to cookies. Available settings function as follows:

* None - Do not add flags to cookies.

* HTTP Only - Add the HTTP Only flag to cookies, which prevents scripts from accessing cookies.

* Secure - Add Secure flag to cookies.

* All - Add both HTTPOnly and Secure flags to cookies.

Possible values: none, httpOnly, secure, all

Default value: AS_ADD_CKI_FLAGS_NONE

fieldConsistencyAction

One or more Form Field Consistency actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldConsistencyaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldConsistencyAction none".

Default value: AS_NONE

CSRFtagAction

One or more Cross-Site Request Forgery (CSRF) Tagging actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -CSRFTagAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -CSRFTagAction none".

Default value: AS_NONE

crossSiteScriptingAction

One or more Cross-Site Scripting (XSS) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -crossSiteScriptingAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -crossSiteScriptingAction none".

Default value: AS_DEFAULT_DISPOSITION

crossSiteScriptingTransformUnsafeHTML

Transform cross-site scripts. This setting configures the application firewall to disable dangerous HTML instead of blocking the request.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cross-site scripting transformations. If it is set to OFF, no cross-site scripting transformations are performed regardless of any other settings.

Possible values: ON, OFF

Default value: OFF

crossSiteScriptingCheckCompleteURLs

Check complete URLs for cross-site scripts, instead of just the query portions of URLs.

Possible values: ON, OFF

Default value: OFF

SQLInjectionAction

One or more HTML SQL Injection actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -SQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -SQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

SQLInjectionTransformSpecialChars

Transform injected SQL code. This setting configures the application firewall to disable SQL special strings instead of blocking the request. Since most SQL servers require a special string to activate an SQL keyword, in most cases a request that contains injected SQL code is safe if special strings are disabled.

CAUTION: Make sure that this parameter is set to ON if you are configuring any SQL injection transformations. If it is set to OFF, no SQL injection transformations are performed regardless of any other settings.

Possible values: ON, OFF

Default value: OFF

SQLInjectionOnlyCheckFieldsWithSQLChars

Check only form fields that contain SQL special strings (characters) for injected SQL code.

Most SQL servers require a special string to activate an SQL request, so SQL code without a special string is harmless to most SQL servers.

Possible values: ON, OFF

Default value: ON

SQLInjectionType

Available SQL injection types.

-SQLSplChar : Checks for SQL Special Chars

-SQLKeyword : Checks for SQL Keywords

-SQLSplCharANDKeyword : Checks for both and blocks if both are found

-SQLSplCharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSplChar, SQLKeyword, SQLSplCharORKeyword, SQLSplCharANDKeyword

Default value: AS_SQLINJECTION_TYPE_CHAR_AND_KEYWORD

SQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

Default value: OFF

fieldFormatAction

One or more Field Format actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of suggested web form fields and field format assignments.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

defaultFieldFormatType

Designate a default field type to be applied to web form fields that do not have a field type explicitly assigned to them.

defaultFieldFormatMinLength

Minimum length, in characters, for data entered into a field that is assigned the default field type.

To disable the minimum and maximum length settings and allow data of any length to be entered into the field, set this parameter to zero (0).

Default value: AS_DEFAULTFIELDFORMAT_DEFAULT_MIN_LEN

Minimum value: 0

Maximum value: 65535

defaultFieldFormatMaxLength

Maximum length, in characters, for data entered into a field that is assigned the default field type.

Default value: AS_DEFAULTFIELDFORMAT_DEFAULT_MAX_LEN

Minimum value: 1

Maximum value: 65535

bufferOverflowAction

One or more Buffer Overflow actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -bufferOverflowAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -bufferOverflowAction none".

Default value: AS_DEFAULT_DISPOSITION

bufferOverflowMaxURLLength

Maximum length, in characters, for URLs on your protected web sites. Requests with longer URLs are blocked.

Default value: AS_BUFFEROVERFLOW_DEFAULT_MAX_URL_LEN

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxHeaderLength

Maximum length, in characters, for HTTP headers in requests sent to your protected web sites. Requests with longer headers are blocked.

Default value: AS_BUFFEROVERFLOW_DEFAULT_MAX_HDR_LEN

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxCookieLength

Maximum length, in characters, for cookies sent to your protected web sites. Requests with longer cookies are blocked.

Default value: AS_BUFFEROVERFLOW_DEFAULT_MAX_COOKIE_LEN

Minimum value: 0

Maximum value: 65535

creditCardAction

One or more Credit Card actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -creditCardAction" followed by the actions to be

enabled. To turn off all actions, type "set appfw profile -creditCardAction none".

Default value: AS_NONE

creditCard

Credit card types that the application firewall should protect.

Default value: AS_CCARD_DEFAULT_CARD_TYPE

creditCardMaxAllowed

Maximum number of credit card numbers that can appear on a web page served by your protected web sites. Pages that contain more credit card numbers are blocked, or the credit card numbers are masked.

Maximum value: 255

creditCardXOut

Mask any credit card number detected in a response by replacing each digit, except the digits in the final group, with the letter "X."

Possible values: ON, OFF

Default value: OFF

requestContentType

Default Content-Type header for requests.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_REQUEST_CONTENT_TYPE

responseContentType

Default Content-Type header for responses.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_RESPONSE_CONTENT_TYPE

XMLDoSAction

One or more XML Denial-of-Service (XDoS) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLDoSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLDoSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLFormatAction

One or more XML Format actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionAction

One or more XML SQL Injection actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionOnlyCheckFieldsWithSQLChars

Check only form fields that contain SQL special characters, which most SQL servers require before accepting an SQL command, for injected SQL.

Possible values: ON, OFF

Default value: ON

XMLSQLInjectionType

Available SQL injection types.

-SQLSplChar : Checks for SQL Special Chars

-SQLKeyword : Checks for SQL Keywords

-SQLSplCharANDKeyword : Checks for both and blocks if both are found

-SQLSplCharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSplChar, SQLKeyword, SQLSplCharORKeyword, SQLSplCharANDKeyword

Default value: AS_SQLINJECTION_TYPE_CHAR_AND_KEYWORD

XMLSQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

Default value: OFF

XMLSQLInjectionParseComments

Parse comments in XML Data and exempt those sections of the request that are from the XML SQL Injection check. You must configure the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

* Check all - Check all content.

* ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.

* Nested - Exempt content that is part of a nested (Microsoft-style) comment.

* ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: AS_CHECKALL

XMLXSSAction

One or more XML Cross-Site Scripting actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLXSSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLXSSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLWSIAction

One or more Web Services Interoperability (WSI) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLWSIAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLWSIAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLAttachmentAction

One or more XML Attachment actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLAttachmentAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLAttachmentAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLValidationAction

One or more XML Validation actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLValidationAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLValidationAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLErrorObject

Name to assign to the XML Error Object, which the application firewall displays when a user request is blocked.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore

characters. Cannot be changed after the XML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my XML error object" or 'my XML error object'\\).

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

customSettings

Object name for custom settings.

This check is applicable to Profile Type: HTML, XML.

Default value: NS_S_AS_CUSTOM_OBJECT_DEFAULT

signatures

Object name for signatures.

This check is applicable to Profile Type: HTML, XML.

Default value: NS_S_AS_CUSTOM_OBJECT_DEFAULT

XMLSOAPFaultAction

One or more XML SOAP Fault Filtering actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

* Remove - Remove all violations for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSOAPFaultAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSOAPFaultAction none".

Default value: AS_DEFAULT_DISPOSITION

useHTMLErrorObject

Send an imported HTML Error object to a user when a request is blocked, instead of redirecting the user to the designated Error URL.

Possible values: ON, OFF

Default value: OFF

errorURL

URL that application firewall uses as the Error URL.

Default value: NS_S_AS_ERROR_URL_DEFAULT

HTMLErrorObject

Name to assign to the HTML Error Object.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the HTML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my HTML error object" or 'my HTML error object'\\).

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

logEveryPolicyHit

Log every profile match, regardless of security checks results.

Possible values: ON, OFF

Default value: OFF

stripComments

Strip HTML comments.

This check is applicable to Profile Type: HTML.

Possible values: ON, OFF

Default value: OFF

stripHtmlComments

Strip HTML comments before forwarding a web page sent by a protected web site in response to a user request.

Possible values: none, all, exclude_script_tag

Default value: AS_STRIP_COMMENT_NONE

stripXmlComments

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: none, all

Default value: AS_STRIP_COMMENT_NONE

exemptClosureURLsFromSecurityChecks

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: ON, OFF

Default value: ON

defaultCharSet

Default character set for protected web pages. Web pages sent by your protected web sites in response to user requests are assigned this character set if the page does not already specify a character set. The character sets supported by the application firewall are:

* iso-8859-1 (English US)

* big5 (Chinese Traditional)

* gb2312 (Chinese Simplified)

* sjis (Japanese Shift-JIS)

* euc-jp (Japanese EUC-JP)

* iso-8859-9 (Turkish)

* utf-8 (Unicode)

* euc-kr (Korean)

Default value: NS_S_AS_CHARSET_DEFAULT

Maximum value: 31

postBodyLimit

Maximum allowed HTTP post body size, in bytes.

Default value: AS_DEFAULT_POSTBODYLIMIT

Maximum value: 1000000000

fileUploadMaxNum

Maximum allowed number of file uploads per form-submission request. The maximum setting (65535) allows an unlimited number of uploads.

Default value: AS_DEFAULT_MAX_FILE_UPLOADS

Maximum value: 65535

canonicalizeHTMLResponse

Perform HTML entity encoding for any special characters in responses sent by your protected web sites.

Possible values: ON, OFF

Default value: ON

enableFormTagging

Enable tagging of web form fields for use by the Form Field Consistency and CSRF Form Tagging checks.

Possible values: ON, OFF

Default value: ON

sessionlessFieldConsistency

Perform sessionless Field Consistency Checks.

Possible values: OFF, ON, postOnly

Default value: AS_OFF

sessionlessURLClosure

Enable session less URL Closure Checks.

This check is applicable to Profile Type: HTML.

Possible values: ON, OFF

Default value: OFF

semicolonFieldSeparator

Allow ';' as a form field separator in URL queries and POST form bodies.

Possible values: ON, OFF

Default value: OFF

excludeFileUploadFromChecks

Exclude uploaded files from Form checks.

Possible values: ON, OFF

Default value: OFF

SQLInjectionParseComments

Parse HTML comments and exempt them from the HTML SQL Injection check. You must specify the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

* Check all - Check all content.

* ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.

* Nested - Exempt content that is part of a nested (Microsoft-style) comment.

* ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: AS_DEFAULT_SQLINJECTIONPARSECOMMENTS

invalidPercentHandling

Configure the method that the application firewall uses to handle percent-encoded names and values. Available settings function as follows:

* apache_mode - Apache format.

* asp_mode - Microsoft ASP format.

* secure_mode - Secure format.

Possible values: apache_mode, asp_mode, secure_mode

Default value: AS_PERCENT_DECODE_SECURE_MODE

type

Application firewall profile type, which controls which security checks and settings are applied to content that is filtered with the profile. Available settings function as follows:

* HTML - HTML-based web sites.

* XML - XML-based web sites and services.

* HTML XML (Web 2.0) - Sites that contain both HTML and XML content, such as ATOM feeds, blogs, and RSS feeds.

Default value: AF_PROFILE_TYPE_HTML

checkRequestHeaders

Check request headers as well as web forms for injected SQL and cross-site scripts.

Possible values: ON, OFF

Default value: OFF

comment

Any comments about the purpose of profile, or other useful information about the profile.

---

rm appfw profile

Removes the specified application firewall profile.

Synopsys

rm appfw profile <name>

Arguments

name

Name of the profile.

set appfw profile

Modifies the specified parameters of the specified application firewall profile.

Synopsys

set appfw profile <name> [-startURLAction <startURLAction> ...] [-contentTypeAction <contentTypeAction> ...] [-startURLClosure (ON | OFF)] [-denyURLAction <denyURLAction> ...] [-RefererHeaderCheck <RefererHeaderCheck>] [-cookieConsistencyAction <cookieConsistencyAction> ...] [-cookieTransforms (ON | OFF)] [-cookieEncryption <cookieEncryption>] [-cookieProxying (none | sessionOnly)] [-addCookieFlags <addCookieFlags>] [-fieldConsistencyAction <fieldConsistencyAction> ...] [-CSRFtagAction <CSRFtagAction> ...] [-crossSiteScriptingAction <crossSiteScriptingAction> ...] [-crossSiteScriptingTransformUnsafeHTML (ON | OFF)] [-crossSiteScriptingCheckCompleteURLs (ON | OFF)] [-SQLInjectionAction <SQLInjectionAction> ...] [-SQLInjectionTransformSpecialChars (ON | OFF)] [-SQLInjectionType <SQLInjectionType>] [-SQLInjectionCheckSQLWildChars (ON | OFF)] [-fieldFormatAction <fieldFormatAction> ...] [-defaultFieldFormatType <string>] [-defaultFieldFormatMinLength <positive_integer>] [-defaultFieldFormatMaxLength <positive_integer>] [-bufferOverflowAction <bufferOverflowAction> ...] [-bufferOverflowMaxURLLength <positive_integer>] [-bufferOverflowMaxHeaderLength <positive_integer>] [-bufferOverflowMaxCookieLength <positive_integer>] [-creditCardAction <creditCardAction> ...] [-creditCard <creditCard> ...] [-creditCardMaxAllowed <positive_integer>] [-creditCardXOut (ON | OFF)] [-requestContentType <string>] [-responseContentType <string>] [-XMLDoSAction <XMLDoSAction> ...] [-XMLFormatAction <XMLFormatAction> ...] [-XMLSQLInjectionAction <XMLSQLInjectionAction> ...] [-XMLSQLInjectionType <XMLSQLInjectionType>] [-XMLSQLInjectionCheckSQLWildChars (ON | OFF)] [-XMLSQLInjectionParseComments <XMLSQLInjectionParseComments>] [-XMLXSSAction <XMLXSSAction> ...] [-XMLWSIAction <XMLWSIAction> ...] [-XMLAttachmentAction <XMLAttachmentAction> ...] [-XMLValidationAction <XMLValidationAction> ...] [-XMLErrorObject <string>] [-signatures <string>] [-XMLSOAPFaultAction <XMLSOAPFaultAction> ...] [-useHTMLErrorObject (ON | OFF)] [-errorURL <expression>] [-HTMLErrorObject <string>] [-logEveryPolicyHit (ON | OFF)] [-stripHtmlComments <stripHtmlComments>] [-stripXmlComments (none | all)] [-exemptClosureURLsFromSecurityChecks (ON | OFF)] [-defaultCharSet <string>] [-postBodyLimit <positive_integer>] [-fileUploadMaxNum <positive_integer>] [-canonicalizeHTMLResponse (ON | OFF)] [-enableFormTagging (ON | OFF)] [-sessionlessFieldConsistency <sessionlessFieldConsistency>] [-sessionlessURLClosure (ON | OFF)] [-semicolonFieldSeparator (ON | OFF)] [-excludeFileUploadFromChecks (ON | OFF)] [-SQLInjectionParseComments <SQLInjectionParseComments>] [-invalidPercentHandling <invalidPercentHandling>] [-type (HTML | XML) ...] [-checkRequestHeaders (ON | OFF)] [-comment <string>]

Arguments

name

Name of the profile that you want to modify.

startURLAction

One or more Start URL actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -startURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -startURLaction none".

Default value: AS_DEFAULT_DISPOSITION

contentTypeAction

One or more Content-type actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -contentTypeaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -contentTypeaction none".

Default value: AS_DEFAULT_CONTENT_TYPE_DISPOSITION

startURLClosure

Toggle the state of Start URL Closure.

Possible values: ON, OFF

Default value: OFF

denyURLAction

One or more Deny URL actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

NOTE: The Deny URL check takes precedence over the Start URL check. If you enable blocking for the Deny URL check, the application firewall blocks any URL that is explicitly blocked by a Deny URL, even if the same URL would otherwise be allowed by the Start URL check.

CLI users: To enable one or more actions, type "set appfw profile -denyURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -denyURLaction none".

Default value: AS_DEFAULT_DISPOSITION

RefererHeaderCheck

Enable validation of Referer headers.

Referer validation ensures that a web form that a user sends to your web site originally came from your web site, not an outside attacker.

Although this parameter is part of the Start URL check, referer validation protects against cross-site request forgery (CSRF) attacks, not Start URL attacks.

Possible values: OFF, if_present, AlwaysExceptStartURLs, AlwaysExceptFirstRequest

Default value: AS_HEADER_CHECK_OFF

cookieConsistencyAction

One or more Cookie Consistency actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -cookieConsistencyAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -cookieConsistencyAction none".

Default value: AS_NONE

cookieTransforms

Perform the specified type of cookie transformation.

Available settings function as follows:

* Encryption - Encrypt cookies.

* Proxying - Mask contents of server cookies by sending proxy cookie to users.

* Cookie flags - Flag cookies as HTTP only to prevent scripts on user's browser from accessing and possibly modifying them.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cookie transformations. If it is set to OFF, no cookie transformations are performed regardless of any other settings.

Possible values: ON, OFF

cookieEncryption

Type of cookie encryption. Available settings function as follows:

* None - Do not encrypt cookies.

* Decrypt Only - Decrypt encrypted cookies, but do not encrypt cookies.

* Encrypt Session Only - Encrypt session cookies, but not permanent cookies.

* Encrypt All - Encrypt all cookies.

Possible values: none, decryptOnly, encryptSessionOnly, encryptAll

Default value: AS_CKI_ENCRYPT_NONE

cookieProxying

Cookie proxy setting. Available settings function as follows:

* None - Do not proxy cookies.

* Session Only - Proxy session cookies by using the NetScaler session ID, but do not proxy permanent cookies.

Possible values: none, sessionOnly

Default value: AS_CKI_PROXY_NONE

addCookieFlags

Add HttpOnly and Secure flags to cookies

Possible values: none, httpOnly, secure, all

Default value: AS_ADD_CKI_FLAGS_NONE

fieldConsistencyAction

One or more Form Field Consistency actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldConsistencyaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldConsistencyAction none".

Default value: AS_NONE

CSRFtagAction

One or more Cross-Site Request Forgery (CSRF) Tagging actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -CSRFTagAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -CSRFTagAction none".

Default value: AS_NONE

crossSiteScriptingAction

One or more Cross-Site Scripting (XSS) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -crossSiteScriptingAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -crossSiteScriptingAction none".

Default value: AS_DEFAULT_DISPOSITION

crossSiteScriptingTransformUnsafeHTML

Transform cross-site scripts. This setting configures the application firewall to disable dangerous HTML instead of blocking the request.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cross-site scripting transformations. If it is set to OFF, no cross-site scripting transformations are performed regardless of any other settings.

Possible values: ON, OFF

crossSiteScriptingCheckCompleteURLs

Check complete URLs for cross-site scripts, instead of just the query portions of URLs.

Possible values: ON, OFF

SQLInjectionAction

One or more HTML SQL Injection actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -SQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -SQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

SQLInjectionTransformSpecialChars

Transform injected SQL code. This setting configures the application firewall to disable SQL special strings instead of blocking the request. Since most SQL servers require a special string to activate an SQL keyword, in most cases a request that contains injected SQL code is safe if special strings are disabled.

CAUTION: Make sure that this parameter is set to ON if you are configuring any SQL injection transformations. If it is set to OFF, no SQL injection transformations are performed regardless of any other settings.

Possible values: ON, OFF

SQLInjectionOnlyCheckFieldsWithSQLChars

Check only form fields that contain SQL special strings (characters) for injected SQL code.

Most SQL servers require a special string to activate an SQL request, so SQL code without a special string is harmless to most SQL servers.

Possible values: ON, OFF

SQLInjectionType

Available SQL injection types.

-SQLSplChar : Checks for SQL Special Chars

-SQLKeyword : Checks for SQL Keywords

-SQLSplCharANDKeyword : Checks for both and blocks if both are found

-SQLSplCharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSplChar, SQLKeyword, SQLSplCharORKeyword, SQLSplCharANDKeyword

SQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

fieldFormatAction

One or more Field Format actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of suggested web form fields and field format assignments.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

defaultFieldFormatType

Designate a default field type to be applied to web form fields that do not have a field type explicitly assigned to them.

defaultFieldFormatMinLength

Minimum length, in characters, for data entered into a field that is assigned the default field type.

To disable the minimum and maximum length settings and allow data of any length to be entered into the field, set this parameter to zero (0).

Default value: AS_DEFAULTFIELDFORMAT_DEFAULT_MIN_LEN

Minimum value: 0

Maximum value: 65535

defaultFieldFormatMaxLength

Maximum length, in characters, for data entered into a field that is assigned the default field type.

Default value: AS_DEFAULTFIELDFORMAT_DEFAULT_MAX_LEN

Minimum value: 1

Maximum value: 65535

bufferOverflowAction

One or more Buffer Overflow actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -bufferOverflowAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -bufferOverflowAction none".

Default value: AS_DEFAULT_DISPOSITION

bufferOverflowMaxURLLength

Maximum length, in characters, for URLs on your protected web sites. Requests with longer URLs are blocked.

Default value: AS_BUFFEROVERFLOW_DEFAULT_MAX_URL_LEN

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxHeaderLength

Maximum length, in characters, for HTTP headers in requests sent to your protected web sites. Requests with longer headers are blocked.

Default value: AS_BUFFEROVERFLOW_DEFAULT_MAX_HDR_LEN

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxCookieLength

Maximum length, in characters, for cookies sent to your protected web sites. Requests with longer cookies are blocked.

Default value: AS_BUFFEROVERFLOW_DEFAULT_MAX_COOKIE_LEN

Minimum value: 0

Maximum value: 65535

creditCardAction

One or more Credit Card actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -creditCardAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -creditCardAction none".

Default value: AS_NONE

creditCard

Credit card types that the application firewall should protect.

Default value: AS_CCARD_DEFAULT_CARD_TYPE

creditCardMaxAllowed

Maximum number of credit card numbers that can appear on a web page served by your protected web sites. Pages that contain more credit card numbers are blocked, or the credit card numbers are masked.

Maximum value: 255

creditCardXOut

Mask any credit card number detected in a response by replacing each digit, except the digits in the final group, with the letter "X."

Possible values: ON, OFF

requestContentType

Default Content-Type header for requests.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_REQUEST_CONTENT_TYPE

responseContentType

Default Content-Type header for responses.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_RESPONSE_CONTENT_TYPE

XMLDoSAction

One or more XML Denial-of-Service (XDoS) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLDoSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLDoSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLFormatAction

One or more XML Format actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionAction

One or more XML SQL Injection actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionOnlyCheckFieldsWithSQLChars

Check only form fields that contain SQL special characters, which most SQL servers require before accepting an SQL command, for injected SQL.

Possible values: ON, OFF

XMLSQLInjectionType

Available SQL injection types.

-SQLSplChar : Checks for SQL Special Chars

-SQLKeyword : Checks for SQL Keywords

-SQLSplCharANDKeyword : Checks for both and blocks if both are found

-SQLSplCharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSplChar, SQLKeyword, SQLSplCharORKeyword, SQLSplCharANDKeyword

XMLSQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

XMLSQLInjectionParseComments

Parse comments in XML Data and exempt those sections of the request that are from the XML SQL Injection check. You must configure the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

* Check all - Check all content.

* ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.

* Nested - Exempt content that is part of a nested (Microsoft-style) comment.

* ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: AS_CHECKALL

XMLXSSAction

One or more XML Cross-Site Scripting actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLXSSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLXSSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLWSIAction

One or more Web Services Interoperability (WSI) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLWSIAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLWSIAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLAttachmentAction

One or more XML Attachment actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of exceptions to this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLAttachmentAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLAttachmentAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLValidationAction

One or more XML Validation actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLValidationAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLValidationAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLErrorObject

Name to assign to the XML Error Object, which the application firewall displays when a user request is blocked.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the XML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my XML error object" or 'my XML error object'\\).

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

customSettings

Object name for custom settings.

This check is applicable to Profile Type: HTML, XML.

Default value: NS_S_AS_CUSTOM_OBJECT_DEFAULT

signatures

Object name for signatures.

This check is applicable to Profile Type: HTML, XML.

Default value: NS_S_AS_CUSTOM_OBJECT_DEFAULT

XMLSOAPFaultAction

One or more XML SOAP Fault Filtering actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

* Remove - Remove all violations for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSOAPFaultAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSOAPFaultAction none".

Default value: AS_DEFAULT_DISPOSITION

useHTMLErrorObject

Send an imported HTML Error object to a user when a request is blocked, instead of redirecting the user to the designated Error URL.

Possible values: ON, OFF

errorURL

URL that application firewall uses as the Error URL.

Default value: NS_S_AS_ERROR_URL_DEFAULT

HTMLErrorObject

Name to assign to the HTML Error Object.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the HTML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my HTML error object" or 'my HTML error object'\\).

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

logEveryPolicyHit

Log every profile match, regardless of security checks results.

Possible values: ON, OFF

stripComments

Strip HTML comments.

This check is applicable to Profile Type: HTML.

Possible values: ON, OFF

stripHtmlComments

Strip HTML comments before forwarding a web page sent by a protected web site in response to a user request.

Possible values: none, all, exclude_script_tag

stripXmlComments

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: none, all

exemptClosureURLsFromSecurityChecks

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: ON, OFF

defaultCharSet

Default character set for protected web pages. Web pages sent by your protected web sites in response to user requests are assigned this character set if the page does not already specify a character set. The character sets supported by the application firewall are:

* iso-8859-1 (English US)

* big5 (Chinese Traditional)

* gb2312 (Chinese Simplified)

* sjis (Japanese Shift-JIS)

* euc-jp (Japanese EUC-JP)

* iso-8859-9 (Turkish)

* utf-8 (Unicode)

* euc-kr (Korean)

Default value: NS_S_AS_CHARSET_DEFAULT

Maximum value: 31

postBodyLimit

Maximum allowed HTTP post body size, in bytes.

Default value: AS_DEFAULT_POSTBODYLIMIT

Maximum value: 1000000000

fileUploadMaxNum

Maximum allowed number of file uploads per form-submission request. The maximum setting (65535) allows an unlimited number of uploads.

Default value: AS_DEFAULT_MAX_FILE_UPLOADS

Maximum value: 65535

canonicalizeHTMLResponse

Perform HTML entity encoding for any special characters in responses sent by your protected web sites.

Possible values: ON, OFF

Default value: ON

enableFormTagging

Enable tagging of web form fields for use by the Form Field Consistency and CSRF Form Tagging checks.

Possible values: ON, OFF

Default value: ON

sessionlessFieldConsistency

Perform sessionless Field Consistency Checks.

Possible values: OFF, ON, postOnly

Default value: AS_OFF

sessionlessURLClosure

Enable session less URL Closure Checks.

This check is applicable to Profile Type: HTML.

Possible values: ON, OFF

Default value: OFF

semicolonFieldSeparator

Allow ';' as a form field separator in URL queries and POST form bodies.

Possible values: ON, OFF

Default value: OFF

excludeFileUploadFromChecks

Exclude uploaded files from Form checks.

Possible values: ON, OFF

Default value: OFF

SQLInjectionParseComments

Parse HTML comments and exempt them from the HTML SQL Injection check. You must specify the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

* Check all - Check all content.

* ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.

* Nested - Exempt content that is part of a nested (Microsoft-style) comment.

* ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: AS_DEFAULT_SQLINJECTIONPARSECOMMENTS

invalidPercentHandling

Configure the method that the application firewall uses to handle percent-encoded names and values. Available settings function as follows:

* apache_mode - Apache format.

* asp_mode - Microsoft ASP format.

* secure_mode - Secure format.

Possible values: apache_mode, asp_mode, secure_mode

Default value: AS_PERCENT_DECODE_SECURE_MODE

type

Application firewall profile type, which controls which security checks and settings are applied to content that is filtered with the profile. Available settings function as follows:

* HTML - HTML-based web sites.

* XML - XML-based web sites and services.

* HTML XML (Web 2.0) - Sites that contain both HTML and XML content, such as ATOM feeds, blogs, and RSS feeds.

Default value: AF_PROFILE_TYPE_HTML

checkRequestHeaders

Check request headers as well as web forms for injected SQL and cross-site scripts.

Possible values: ON, OFF

Default value: OFF

comment

Any comments about the purpose of profile, or other useful information about the profile.

---

unset appfw profile

Use this command to remove appfw profile settings.Refer to the set appfw profile command for meanings of the arguments.

Synopsys

unset appfw profile <name> [-startURLAction] [-contentTypeAction] [-startURLClosure] [-denyURLAction] [-RefererHeaderCheck] [-cookieConsistencyAction] [-cookieTransforms] [-cookieEncryption] [-cookieProxying] [-addCookieFlags] [-fieldConsistencyAction] [-CSRFtagAction] [-crossSiteScriptingAction] [-crossSiteScriptingTransformUnsafeHTML] [-crossSiteScriptingCheckCompleteURLs] [-SQLInjectionAction] [-SQLInjectionTransformSpecialChars] [-SQLInjectionType] [-SQLInjectionCheckSQLWildChars] [-fieldFormatAction] [-defaultFieldFormatType] [-defaultFieldFormatMinLength] [-defaultFieldFormatMaxLength] [-bufferOverflowAction] [-bufferOverflowMaxURLLength] [-bufferOverflowMaxHeaderLength] [-bufferOverflowMaxCookieLength] [-creditCardAction] [-creditCard] [-creditCardMaxAllowed] [-creditCardXOut] [-requestContentType] [-responseContentType] [-XMLDoSAction] [-XMLFormatAction] [-XMLSQLInjectionAction] [-XMLSQLInjectionType] [-XMLSQLInjectionCheckSQLWildChars] [-XMLSQLInjectionParseComments] [-XMLXSSAction] [-XMLWSIAction] [-XMLAttachmentAction] [-XMLValidationAction] [-XMLErrorObject] [-signatures] [-XMLSOAPFaultAction] [-useHTMLErrorObject] [-errorURL] [-HTMLErrorObject] [-logEveryPolicyHit] [-stripHtmlComments] [-stripXmlComments] [-exemptClosureURLsFromSecurityChecks] [-defaultCharSet] [-postBodyLimit] [-fileUploadMaxNum] [-canonicalizeHTMLResponse] [-enableFormTagging] [-sessionlessFieldConsistency] [-sessionlessURLClosure] [-semicolonFieldSeparator] [-excludeFileUploadFromChecks] [-SQLInjectionParseComments] [-invalidPercentHandling] [-type] [-checkRequestHeaders] [-comment]

---

bind appfw profile

Binds the specified exemption (relaxation) or rule to the specified application firewall profile. NOTE: You should not attempt to bind more than one exemption or rule at a time by using this command.

Synopsys

bind appfw profile <name> (-startURL <expression> | -denyURL <expression> | (-fieldConsistency <string> <formActionURL> [-isRegex (REGEX | NOTREGEX)]) | (-cookieConsistency <string> [-isRegex (REGEX | NOTREGEX)]) | (-SQLInjection <string> <formActionURL> [-isRegex (REGEX | NOTREGEX)] [-location <location>]) | (-CSRFTag <expression> <CSRFFormActionURL>) | (-crossSiteScripting <string> <formActionURL> [-isRegex (REGEX | NOTREGEX)] [-location <location>]) | (-fieldFormat <string> <formActionURL> <fieldType> [-fieldFormatMinLength <positive_integer>] [-fieldFormatMaxLength <positive_integer>] [-isRegex (REGEX | NOTREGEX)]) | (-safeObject <string> <expression> <maxMatchLength> [-action <action> ...]) | -trustedLearningClients <ip_addr[/prefix]|ipv6_addr[/prefix]|*> | (-XMLDoSURL <expression> [-XMLMaxElementDepthCheck (ON | OFF) [-XMLMaxElementDepth <positive_integer>]] [-XMLMaxElementNameLengthCheck (ON | OFF) [-XMLMaxElementNameLength <positive_integer>]] [-XMLMaxElementsCheck (ON | OFF) [-XMLMaxElements <positive_integer>]] [-XMLMaxElementChildrenCheck (ON | OFF) [-XMLMaxElementChildren <positive_integer>]] [-XMLMaxAttributesCheck (ON | OFF) [-XMLMaxAttributes <positive_integer>]] [-XMLMaxAttributeNameLengthCheck (ON | OFF) [-XMLMaxAttributeNameLength <positive_integer>]] [-XMLMaxAttributeValueLengthCheck (ON | OFF) [-XMLMaxAttributeValueLength <positive_integer>]] [-XMLMaxCharDATALengthCheck (ON | OFF) [-XMLMaxCharDATALength <positive_integer>]] [-XMLMaxFileSizeCheck (ON | OFF) [-XMLMaxFileSize <positive_integer>]] [-XMLMinFileSizeCheck (ON | OFF) [-XMLMinFileSize <positive_integer>]] [-XMLBlockPI (ON | OFF)] [-XMLBlockDTD (ON | OFF)] [-XMLBlockExternalEntities (ON | OFF)] [-XMLMaxEntityExpansionsCheck (ON | OFF) [-XMLMaxEntityExpansions <positive_integer>]] [-XMLMaxEntityExpansionDepthCheck (ON | OFF) [-XMLMaxEntityExpansionDepth <positive_integer>]] [-XMLMaxNamespacesCheck (ON | OFF) [-XMLMaxNamespaces <positive_integer>]] [-XMLMaxNamespaceUriLengthCheck (ON | OFF) [-XMLMaxNamespaceUriLength <positive_integer>]] [-XMLSOAPArrayCheck (ON | OFF) [-XMLMaxSOAPArraySize <positive_integer>] [-XMLMaxSOAPArrayRank <positive_integer>]]) | (-XMLWSIURL <expression> [-XMLWSIChecks <string>]) | (-XMLValidationURL <expression> (-XMLRequestSchema <string> | (-XMLWSDL <string> [-XMLAdditionalSOAPHeaders (ON | OFF)] [-XMLEndPointCheck (ABSOLUTE | RELATIVE)]) | -XMLValidateSOAPEnvelope (ON | OFF)) [-XMLResponseSchema <string>] [-XMLValidateResponse (ON | OFF)]) | (-XMLAttachmentURL <expression> [-XMLMaxAttachmentSizeCheck (ON | OFF) [-XMLMaxAttachmentSize <positive_integer>]] [-XMLAttachmentContentTypeCheck (ON | OFF) [-XMLAttachmentContentType <expression>]]) | (-XMLSQLInjection <string> [-isRegex (REGEX | NOTREGEX)] [-location (ELEMENT | ATTRIBUTE)]) | (-XMLXSS <string> [-isRegex (REGEX | NOTREGEX)] [-location (ELEMENT | ATTRIBUTE)]) | -contentType <expression> | -excludeResContentType <expression>) [-comment <string>] [-state (ENABLED | DISABLED)]

Arguments

name

Name of the profile to which to bind an exemption or rule.

startURL

Add the specified URL to the start URL list.

Enclose URLs in double quotes to ensure preservation of any embedded spaces or non-alphanumeric characters.

denyURL

Add the specified URL to the deny URL list.

Enclose URLs in double quotes to ensure preservation of any embedded spaces or non-alphanumeric characters.

fieldConsistency

Exempt the specified web form field and form action URL from the form field consistency check, or exempt the specified cookie from the cookie consistency check.

A form field consistency exemption (relaxation) consists of the following items:

* Web form field name. Name of the form field to exempt from this check.

* Form action URL. Action URL for the web form.

* IsRegex flag. The IsRegex flag, followed by YES if the form action URL is a regular expression, or NO if it is a literal string.

cookieConsistency

A cookie consistency exemption (relaxation) consists of the following items:

* Cookie name. Name of the cookie to exempt from this check.

* IsRegex flag. The IsRegex flag, followed by YES if the cookie name is a regular expression, or NO if it is a literal string.

SQLInjection

Exempt the specified HTTP header, web form field and the form action URL, or cookie from the SQL injection check.

An SQL injection exemption (relaxation) consists of the following items:

*Item name. Name of the web form field, cookie, or HTTP header to exempt from this check.

* Form action URL. If the item to be exempted is a web form field, the action URL for the web form.

* IsRegex flag. The IsRegex flag, followed by YES if the name or form action URL is a regular expression, or NO if it is a literal string.

* Location. Location that should be examined by the SQL injection check, either FORMFIELD for web form field, HEADER for HTTP header, or COOKIE for cookie.

CSRFTag

Exempt the specified form field and web form from the cross-site request forgery (CSRF tagging) check.

A CSRF tagging exemption (relaxation) consists of the following items:

* Web form field name. Regular expression that describes the web form field to exempt from this check.

* Form action URL. The action URL for the web form.

crossSiteScripting

Exempt the specified string, found in the specified HTTP header, cookie, or web form, from the cross-site scripting check.

A cross-site scripting check exemption (relaxation) consists of the following items:

* HTML to exempt. The string to exempt from the cross-site scripting check.

* URL. The URL to exempt.

* IsRegex flag. The IsRegex flag, followed by YES if the URL is a regular expression, or NO if it is a literal string.

* location. Location which should be examined by the cross-site scripting check, either FORMFIELD for web form field, HEADER for HTTP header, or COOKIE for cookie.

fieldFormat

Impose the specified format on content returned by users in the specified web form field.

A field format rule consists of the following items:

* Form field name. The name of the form field.

* Form action URL. The form action URL for the web form.

* Field type. The field type (format) to enforce on the specified web form field.

* Field format minimum length. The minimum length allowed for data in the specified field. If 0, field can be left blank.

* Field format maximum length. The maximum length allowed for data in the specified field.

* IsRegex flag. The IsRegex flag, followed by YES if the URL is a regular expression, or NO if it is a literal string.

safeObject

Protect web sites from exposing sensitive private information such as social security numbers, credit card numbers, driver's license numbers, passport numbers, and any other type of private information that can be described by a regular expression.

A safe object consists of the following items:

* Name. A name that describes the type of information that the safe object is to protect.

* Expression. PCRE-format regular expression that describes the information to be protected.

* Maximum match length. Maximum length of a matched string.

* Action. "X-Out" to mask blocked information with the letter X, or "Remove" to remove the information.

trustedLearningClients

Trusted host/network learning IP.

This binding is appilicable to profile Type: HTML, XML.

comment

Any comments about the purpose of profile, or other useful information about the profile.

state

Enabled.

Possible values: ENABLED, DISABLED

Default value: ENABLED

XMLDoSURL

Exempt the specified URL from the specified XML denial-of-service (XDoS) attack protections.

An XDoS exemption (relaxation) consists of the following items:

* URL. PCRE-format regular expression for the URL or URLs to be exempted.

* Maximum-element-depth-check toggle. ON to enable this check, OFF to disable it.

* Maximum-element-depth-check toggle. ON to enable, OFF to disable.

* Maximum-element-depth-check level. Positive integer representing the maximum allowed depth of nested XML elements.

* Maximum-element-name-length-check toggle. ON to enable, OFF to disable.

* Maximum element name length. Positive integer representing the maximum allowed length of XML element names.

* Maximum-number-of-elements-check toggle. ON to enable, OFF to disable.

* Maximum number of elements. Positive integer representing the maximum allowed number of XML elements.

* Maximum-number-of-element-children-check toggle. ON to enable, OFF to disable.

* Maximum number of element children. Positive integer representing the maximum allowed number of XML element children.

* Maximum-number-of-attributes-check toggle. ON to enable, OFF to disable.

* Maximum number of attributes. Positive integer representing the maximum allowed number of XML attributes.

* Maximum-attribute-name-length-check toggle. ON to enable, OFF to disable.

* Maximum attribute name length. Positive integer representing the maximum allowed length of XML attribute names.

* Maximum-attribute-value-length-check toggle. ON to enable, OFF to disable.

* Maximum attribute value length. Positive integer representing the maximum allowed length of XML attribute values.

* Maximum-character-data-length-check toggle. ON to enable, OFF to disable.

* Maximum character-data length. Positive integer representing the maximum allowed length of XML character data.

* Maximum-file-size-check toggle. ON to enable, OFF to disable.

* Maximum file size. Positive integer representing the maximum allowed size, in bytes. of attached or uploaded files.

* Minimum-file-size-check toggle. ON to enable, OFF to disable.

* Minimum file size. Positive integer representing the minimum allowed size, in bytes, of attached or uploaded files.

* Maximum-number-of-entity-expansions-check toggle. ON to enable, OFF to disable.

* Maximum number of entity expansions. Positive integer representing the maximum allowed number of XML entity expansions.

* Maximum-number-of XML-namespaces-check toggle. ON to enable, OFF to disable.

* Maximum number of XML namespaces. Positive integer representing the maximum allowed number of XML namespaces.

* Maximum-XML-namespace-URI-length-check toggle. ON to enable, OFF to disable.

* MaximumXML-namespace URI length. Positive integer representing the maximum allowed length of XML namespace URIs.

* Block-processing-instructions toggle. Block XML processing instructions. ON to enable, OFF to disable.

* Block-DTD toggle. Block design type documents (DTDs). ON to enable, OFF to disable.

* Block-external-XML-entitites toggle. ON to enable, OFF to disable.

* Maximum-SOAP-array-check toggle. ON to enable, OFF to disable.

* Maximum SOAP-array size. Positive integer representing the maximum allowed size of XML SOAP arrays.

* Maximum SOAP-array rank. Positive integer representing the maximum rank (dimensions) of any single XML SOAP array.

XMLWSIURL

Exempt the specified URL from the web services interoperability (WS-I) check. The URL is specified as a PCRE-format regular expression, which can match one or more URLs.

XMLValidationURL

Exempt the specified URL from the XML message validation check.

An XML message validation exemption (relaxation) consists of the following items:

* URL. PCRE-format regular expression that matches the URL(s) to be exempted.

* XML-request-schema toggle. Use the specified XML schema to validate requests. ON to enable, OFF to disable.

* XML request schema. XML schema to use for validating requests.

* XML-response-schema toggle. Use the specified XML schema to validate responses. ON to enable, OFF to disable.

* XML response schema. XML schema to use for validating responses.

* WSDL toggle. Use the specified WSDL to validate. ON to enable, OFF to disable.

* WSDL. WSDL to use for validation.

* SOAP-envelope toggle. Validate against the SOAP envelope. ON to enable, OFF to disable.

* Additional-SOAP-headers toggle. Validate against the extended list of SOAP headers. ON to enable, OFF to disable.

* XML-end-point check. ABSOLUTE to use an absolute end point, RELATIVE to use a relative end point.

XMLAttachmentURL

Exempt the specified URL from the XML attachment check.

An XML attachment exemption (relaxation) consists of the following items:

* URL. PCRE-format regular expression that matches the URL(s) to be exempted.

* Maximum-attachment-size-check toggle. ON to enable, OFF to disable.

* Maximum attachment size. Positive integer representing the maximum allowed size in bytes for each XML attachment.

* Attachment-content-type-check toggle. ON to enable, OFF to disable.

* Attachment content type. PCRE-format regular expression that specifies the list of MIME content types allowed for XML attachments.

XMLSQLInjection

Exempt the specified URL from the XML SQL injection check.

An XML attachment exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

XMLXSS

Exempt the specified URL from the XML cross-site scripting (XSS) check.

An XML cross-site scripting exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

contentType

Add the specified content-type to the content-type list.Enclose content-type in double quotes to ensure

preservation of any embedded spaces or non-alphanumeric characters.

excludeResContentType

Add the specified content-type to the response content-type list that are to be excluded from inspection. Enclose content-type in double quotes to ensure preservation

of any embedded spaces or non-alphanumeric characters.

---

unbind appfw profile

Unbinds the specified exemption (relaxation) or rule from the specified application firewall profile. See the bind appfw profile command for a description of the parameters.

Synopsys

unbind appfw profile <name> (-startURL <expression> | -denyURL <expression> | (-fieldConsistency <string> <formActionURL>) | -cookieConsistency <string> | (-SQLInjection <string> <formActionURL> [-location <location>]) | (-CSRFTag <string> <CSRFFormActionURL>) | (-crossSiteScripting <string> <formActionURL> [-location <location>]) | (-fieldFormat <string> <formActionURL>) | -safeObject <string> | -trustedLearningClients <ip_addr[/prefix]|ipv6_addr[/prefix]|*> | -XMLDoSURL <expression> | -XMLWSIURL <expression> | -XMLValidationURL <expression> | -XMLAttachmentURL <expression> | (-XMLSQLInjection <string> [-location (ELEMENT | ATTRIBUTE)]) | (-XMLXSS <string> [-location (ELEMENT | ATTRIBUTE)]) | -contentType <expression> | -excludeResContentType <expression>)

Arguments

name

Name of the exemption (relaxation) or rule that you want to unbind.

startURL

Start URL regular expression.

denyURL

Deny URL regular expression.

fieldConsistency

Form field name.

cookieConsistency

Cookie name.

SQLInjection

Form field, header or cookie name.

CSRFTag

CSRF Form origin URL.

This binding is applicable to Profile Type: HTML.

crossSiteScripting

Form field, header or cookie name.

fieldFormat

Field format name.

safeObject

Safe Object name.

trustedLearningClients

Trusted learning Clients IP

XMLDoSURL

XML DoS URL regular expression.

XMLWSIURL

XML WS-I URL regular expression.

XMLValidationURL

XML Message URL regular expression.

XMLAttachmentURL

XML Attachment URL regular expression.

XMLSQLInjection

Exempt the specified URL from the XML SQL injection check.

An XML attachment exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

XMLXSS

Exempt the specified URL from the XML cross-site scripting (XSS) check.

An XML cross-site scripting exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

contentType

content-type regular expression.

excludeResContentType

Response content type regular expression that are to be excluded from inspection.

---

show appfw profile

Displays details of the specified application firewall profile. If no profile is specified, displays a list of all application firewall profiles on the NetScaler appliance.

Synopsys

show appfw profile [<name>]

Arguments

name

Name of the application firewall profile.

summary

fullValues

format

level

Outputs

stateflag

type

The profile type of of this Application Firewall profile. If the profile is of the HTML type, only checks relevant to HTML are applied. If the profile is of the XML type, only checks relevent to XML are applied. if the profile is of the Web 2.0 type, then both types of checks are applied.

defaults

Default configuration to apply to the profile. Basic defaults are intended for standard content that requires little further configuration, such as static web site content. Advanced defaults are intended for specialized content that requires significant specialized configuration, such as heavily scripted or dynamic content.

CLI users: When adding an application firewall profile, you can set either the defaults or the type, but not both. To set both options, create the profile by using the add appfw profile command, and then use the set appfw profile command to configure the other option.

useHTMLErrorObject

Send an imported HTML Error object to a user when a request is blocked, instead of redirecting the user to the designated Error URL.

errorURL

The error page for this profile.

HTMLErrorObject

Name to assign to the HTML Error Object.

Must begin with a letter, number, or the underscore character \\(_\\), and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the HTML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my HTML error object" or 'my HTML error object'\\).

logEveryPolicyHit

Log every profile match, regardless of security checks results.

stripComments

Tells the Application Firewall to strip HTML comments from responses before sending them to the user.NOTE: This attribute is deprecated.Replaced by a new command that provides an option to exclude comments inside <script> tag from getting stripped.

stripHtmlComments

Tells the Application Firewall to strip HTML comments from responses before sending them to the user.

stripXmlComments

Tells the Application Firewall to strip XML comments from responses before sending them to the user.

defaultCharSet

The default character set. The character set that the Application Firewall uses

for web pages that do not explicitly set a different character set.

postBodyLimit

The maximum body size for an HTTP POST.

fileUploadMaxNum

Maximum allowed number of file uploads per form-submission request. The maximum setting (65535) allows an unlimited number of uploads.

canonicalizeHTMLResponse

Tells the Application Firewall to convert any non-ASCII characters into HTML entities before sending responses to the user. This is called 'canonicalization' of HTML responses.

enableFormTagging

Enables tagging of web forms for form field Consistency checks.

sessionlessFieldConsistency

Enable session less form field consistency checks.

sessionlessURLClosure

Enable session less URL closure checks.

semicolonFieldSeparator

Allow ';' as a form field separator in URL queries and POST form bodies.

excludeFileUploadFromChecks

Excludes uploaded files from all web form checks.

SQLInjectionParseComments

Canonicalizes SQL Comments in form fields.

checkRequestHeaders

Check request headers as well as web forms for injected SQL and cross-site scripts.

comment

Comments associated with this profile.

startURLAction

Start URL action types. (BLOCK | LEARN | LOG | STATS | NONE)

contentTypeAction

Content-type action types. (BLOCK | LOG | NONE)

startURL

A regular expression that designates a URL on the Start URL list.

startURLClosure

Enable Start URL closure. When enabled, this feature allows users to start their session at a designated start URL, then navigate from that start URL to any URL on a protected web site by clicking a link on another web page on that web site. Otherwise, requests to any URL that is not explicitly allowed are blocked.

denyURLAction

Deny URL action types. (BLOCK | LOG | STATS | NONE)

denyURL

A regular expression that designates a URL on the Deny URL list.

RefererHeaderCheck

Enable validation of Referer headers.

Referer validation ensures that a web form that a user sends to your web site originally came from your web site, not an outside attacker.

Although this parameter is part of the Start URL check, referer validation protects against cross-site request forgery (CSRF) attacks, not Start URL attacks.

CSRFtagAction

Cross-site request forgery tagging action types. (BLOCK | LOG | STATS | NONE)

CSRFTag

The web form originating URL.

CSRFFormActionURL

The web form action URL.

crossSiteScriptingAction

Cross-site scripting action types. (BLOCK | LEARN | LOG | STATS | NONE)

crossSiteScriptingTransformUnsafeHTML

Enables transformation of unsafe HTML into safe HTML before forwarding a request to the web server.

crossSiteScriptingCheckCompleteURLs

Tells the Application Firewall to check complete URLs rather than just the query

portion of URLs for cross-site scripting violations.

crossSiteScripting

The web form field name.

isRegex

Is the XML XSS exempted field name a regular expression?

formActionURL

Action URL of the form field to which a field format will be assigned.

exemptClosureURLsFromSecurityChecks

Tells the Application Firewall to exempt closure URLs from security checks.

location

Location of XSS injection exception - XML Element or Attribute.

SQLInjectionAction

SQL injection action types. (BLOCK | LEARN | LOG | STATS | NONE)

SQLInjectionTransformSpecialChars

Enables transformation of SQL special characters found in web forms into safe equivalents.

SQLInjectionOnlyCheckFieldsWithSQLChars

Tells the Application Firewall to check form fields that contain SQL special characters only, rather than all form fields, for SQL injection violations.NOTE: This attribute is deprecated.The same functionality is added to SQLInjectionType . Set SQLInjectionType to "SQLSplCharANDKeyword" to get the same result

SQLInjectionType

Available SQL Injection types.

SQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

SQLInjection

The web form field name.

invalidPercentHandling

Configure the method that the application firewall uses to handle percent-encoded names and values. Available settings function as follows:

* apache_mode - Apache format.

* asp_mode - Microsoft ASP format.

* secure_mode - Secure format.

fieldConsistencyAction

Form Field Consistency action types. (BLOCK | LEARN | LOG | STATS | NONE)

fieldConsistency

The web form field name.

cookieConsistencyAction

Cookie consistency action types. (BLOCK | LEARN | LOG | STATS | NONE)

cookieConsistency

The name of the cookie to be checked.

cookieTransforms

Perform the specified type of cookie transformation.

Available settings function as follows:

* Encryption - Encrypt cookies.

* Proxying - Mask contents of server cookies by sending proxy cookie to users.

* Cookie flags - Flag cookies as HTTP only to prevent scripts on user's browser from accessing and possibly modifying them.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cookie transformations. If it is set to OFF, no cookie transformations are performed regardless of any other settings.

cookieEncryption

Type of cookie encryption. Available settings function as follows:

* None - Do not encrypt cookies.

* Decrypt Only - Decrypt encrypted cookies, but do not encrypt cookies.

* Encrypt Session Only - Encrypt session cookies, but not permanent cookies.

* Encrypt All - Encrypt all cookies.

cookieProxying

Proxies server cookies using the Application Firewall session

addCookieFlags

Add the specified flags to cookies. Available settings function as follows:

* None - Do not add flags to cookies.

* HTTP Only - Add the HTTP Only flag to cookies, which prevents scripts from accessing cookies.

* Secure - Add Secure flag to cookies.

* All - Add both HTTPOnly and Secure flags to cookies.

bufferOverflowAction

Buffer overflow action types. (BLOCK | LOG | STATS | NONE)

bufferOverflowMaxURLLength

Maximum allowed length for URLs.

bufferOverflowMaxHeaderLength

Maximum allowed length for HTTP headers.

bufferOverflowMaxCookieLength

Maximum allowed length for cookies.

fieldFormatAction

Field format action types. (BLOCK | LEARN | LOG | STATS | NONE)

defaultFieldFormatType

Name of the default field type, the field type that the Application Firewall will assign to a form field when no specific field type is assigned to that particular form field.

defaultFieldFormatMinLength

Default field type minimum length setting.

defaultFieldFormatMaxLength

Default field type maximum length setting.

fieldFormat

Name of the form field to which a field format will be assigned.

fieldType

The field type you are assigning to this form field.

fieldFormatMinLength

The minimum allowed length for data in this form field.

fieldFormatMaxLength

The maximum allowed length for data in this form field.

creditCardAction

Credit Card action types. (BLOCK | LOG | STATS | NONE)

creditCard

Credit card types. (AMEX | DINERSCLUB| DISCOVER | JBC | MASTERCARD | VISA)

creditCardMaxAllowed

Maximum number of times a credit card number may be seen before action is taken.

creditCardXOut

X-out credit card numbers.

safeObject

Name of the Safe Object.

expression

A regular expression that defines the Safe Object.

maxMatchLength

Maximum match length for a Safe Object expression.

action

Safe Object action types. (BLOCK | LOG | STATS | NONE)

requestContentType

Default content-type for request messages.

responseContentType

Default content-type for response messages.

XMLErrorObject

URL for the xml error page

signatures

Signatures for the profile

XMLFormatAction

XML well-formed request action types. (BLOCK | LOG | STATS | NONE)

XMLDoSAction

XML DOS action types. (BLOCK | LEARN | LOG | STATS | NONE)

XMLSQLInjectionAction

XML SQL Injection action types. (BLOCK | LOG | STATS | NONE)

XMLSQLInjectionOnlyCheckFieldsWithSQLChars

XML flag to check only fields with SQL characters.NOTE: This attribute is deprecated.The same functionality is added to SQLInjectionType. Set XMLSQLInjectionType to "SQLSplCharANDKeyword" to get the same result

XMLSQLInjectionType

Available XML SQL Injection types.

XMLSQLInjectionCheckSQLWildChars

XML flag to check for SQL wild chars.

XMLSQLInjectionParseComments

Canonicalize SQL Comments in XML data.

XMLXSSAction

XML cross-site scripting action types. (BLOCK | LOG | STATS | NONE)

XMLWSIAction

XML WSI action types. (BLOCK | LEARN | LOG | STATS | NONE)

XMLAttachmentAction

XML attachment action types. (BLOCK | LEARN | LOG | STATS | NONE)

XMLValidationAction

XML message validation action types. (BLOCK | LOG | STATS | NONE)

XMLSOAPFaultAction

XML SOAP fault filtering action types. (BLOCK | LOG | STATS | REMOVE | NONE)

XMLDoSURL

XML DoS URL regular expression length.

XMLWSIURL

XML WS-I URL regular expression length.

XMLValidationURL

XML Validation URL regular expression.

XMLAttachmentURL

XML attachment URL regular expression length.

XMLSQLInjection

Exempt the specified URL from the XML SQL injection check.

An XML attachment exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

XMLXSS

Exempt the specified URL from the XML cross-site scripting (XSS) check.

An XML cross-site scripting exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

state

Enabled.

XMLMaxElementDepthCheck

State if XML Max element depth check is ON or OFF.

XMLMaxElementDepth

Maximum nesting (depth) of XML elements. This check protects against documents that have excessive hierarchy depths.

XMLMaxElementNameLengthCheck

State if XML Max element name length check is ON or OFF.

XMLMaxElementNameLength

Specify the longest name of any element (including the expanded namespace) to protect against overflow attacks.

XMLMaxElementsCheck

State if XML Max elements check is ON or OFF.

XMLMaxElements

Specify the maximum number of XML elements allowed. Protects against overflow attacks.

XMLMaxElementChildrenCheck

State if XML Max element children check is ON or OFF.

XMLMaxElementChildren

Specify the maximum number of children allowed per XML element. Protects against overflow attacks.

XMLMaxNodesCheck

State if XML Max nodes check is ON or OFF.

XMLMaxNodes

Specify the maximum number of XML nodes. Protects against overflow attacks.

XMLMaxAttributesCheck

State if XML Max attributes check is ON or OFF.

XMLMaxAttributes

Specify maximum number of attributes per XML element. Protects against overflow attacks.

XMLMaxAttributeNameLengthCheck

State if XML Max attribute name length check is ON or OFF.

XMLMaxAttributeNameLength

Specify the longest name of any XML attribute. Protects against overflow attacks.

XMLMaxAttributeValueLengthCheck

State if XML Max atribute value length is ON or OFF.

XMLMaxAttributeValueLength

Specify the longest value of any XML attribute. Protects against overflow attacks.

XMLMaxCharDATALengthCheck

State if XML Max CDATA length check is ON or OFF.

XMLMaxCharDATALength

Specify the maximum size of CDATA. Protects against overflow attacks and large quantities of unparsed data within XML messages.

XMLMaxFileSizeCheck

State if XML Max file size check is ON or OFF.

XMLMaxFileSize

Specify the maximum size of XML messages. Protects against overflow attacks.

XMLMinFileSizeCheck

State if XML Min file size check is ON or OFF.

XMLMinFileSize

Enforces minimum message size.

XMLBlockPI

State if XML Block PI is ON or OFF. Protects resources from denial of service attacks as SOAP messages cannot have processing instructions (PI) in messages.

XMLBlockDTD

State if XML DTD is ON or OFF. Protects against recursive Document Type Declaration (DTD) entity expansion attacks. Also, SOAP messages cannot have DTDs in messages.

XMLBlockExternalEntities

State if XML Block External Entities Check is ON or OFF. Protects against XML External Entity (XXE) attacks that force applications to parse untrusted external entities (sources) in XML documents.

XMLMaxEntityExpansionsCheck

State if XML Max Entity Expansions Check is ON or OFF.

XMLMaxEntityExpansions

Specify maximum allowed number of entity expansions. Protects aganist Entity Expansion Attack.

XMLMaxEntityExpansionDepthCheck

State if XML Max Entity Expansions Depth Check is ON or OFF.

XMLMaxEntityExpansionDepth

Specify maximum entity expansion depth. Protects aganist Entity Expansion Attack.

XMLMaxNamespacesCheck

State if XML Max namespaces check is ON or OFF.

XMLMaxNamespaces

Specify maximum number of active namespaces. Protects against overflow attacks.

XMLMaxNamespaceUriLengthCheck

State if XML Max namespace URI length check is ON or OFF.

XMLMaxNamespaceUriLength

Specify the longest URI of any XML namespace. Protects against overflow attacks.

XMLSOAPArrayCheck

State if XML SOAP Array check is ON or OFF.

XMLMaxSOAPArraySize

XML Max Total SOAP Array Size. Protects against SOAP Array Abuse attack.

XMLMaxSOAPArrayRank

XML Max Individual SOAP Array Rank. This is the dimension of the SOAP array.

XMLWSIChecks

Specify a comma separated list of relevant WS-I rule IDs. (R1140, R1141)

XMLRequestSchema

XML Schema object for request validation .

XMLResponseSchema

XML Schema object for response validation.

XMLWSDL

WSDL object for soap request validation.

XMLAdditionalSOAPHeaders

Allow addtional soap headers.

XMLEndPointCheck

Modifies the behaviour of the Request URL validation w.r.t. the Service URL.

If set to ABSOLUTE, the entire request URL is validated with the entire URL mentioned in Service of the associated WSDL.

eg: Service URL: http://example.org/ExampleService, Request URL: http//example.com/ExampleService would FAIL the validation.

If set to RELAIVE, only the non-hostname part of the request URL is validated against the non-hostname part of the Service URL.

eg: Service URL: http://example.org/ExampleService, Request URL: http//example.com/ExampleService would PASS the validation.

XMLValidateSOAPEnvelope

Validate SOAP Evelope only.

XMLValidateResponse

Validate response message.

XMLMaxAttachmentSizeCheck

State if XML Max attachment size Check is ON or OFF. Protects against XML requests with large attachment data.

XMLMaxAttachmentSize

Specify maximum attachment size.

XMLAttachmentContentTypeCheck

State if XML attachment content-type check is ON or OFF. Protects against XML requests with illegal attachments.

XMLAttachmentContentType

Specify content-type regular expression.

builtin

Indicates that a profile is a built-in entity.

builtinType

Type of built-in profiles

trustedLearningClients

Specify trusted host/network IP

contentType

A regular expression that designates a content-type on the content-types list.

excludeResContentType

A regular expression that represents the content type of the response that are to be excluded from inspection.

devno

count

---

stat appfw profile

Displays statistics for the specified application firewall profile. If no profile is specified, displays abbreviated statistics for all profiles.

Synopsys

stat appfw profile [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the application firewall profile.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

requests (reqs)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

Request Bytes (reqBytes)

Number of bytes transfered for requests

responses (resps)

HTTP/HTTPS responses sent by your protected web

servers via the Application Firewall.

Response Bytes (resBytes)

Number of bytes transfered for responses

aborts

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

redirects (redirect)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

Long Term Ave Response Time (ms) (longAvgRespTimePP)

Average backend response time in milliseconds since reboot

Recent Ave Response Time (ms) (shortAvgRespTimePP)

Average backend response time in milliseconds over the last 7 seconds

start URL (startURL)

Number of Start URL security check violations seen by the Application Firewall.

deny URL (denyURL)

Number of Deny URL security check violations seen by the Application Firewall.

referer header (refererHdr)

Number of Referer Header security check violations seen by the Application Firewall.

buffer overflow (bufovfl)

Number of Buffer Overflow security check violations seen by the Application Firewall.

cookie consistency (cookie)

Number of Cookie Consistency security check violations seen by the Application Firewall.

CSRF form tag (csrf_tag)

Number of Cross Site Request Forgery form tag security check violations seen by the Application Firewall.

HTML Cross-site scripting (xss)

Number of HTML Cross-Site Scripting security check violations seen by the Application Firewall.

HTML SQL injection (sql)

Number of HTML SQL Injection security check violations seen by the Application Firewall.

field format (fieldfmt)

Number of Field Format security check violations seen by the Application Firewall.

field consistency (fieldcon)

Number of Field Consistency security check violations seen by the Application Firewall.

credit card (ccard)

Number of Credit Card security check violations seen by the Application Firewall.

safe object (safeobj)

Number of Safe Object security check violations seen by the Application Firewall.

Signature Violations (sigs)

Number of Signature violations seen by the Application Firewall.

XML Format (wfcViolations)

Number of XML Format security check violations seen by the Application Firewall.

XML Denial of Service (XDoS) (xdosViolations)

Number of XML Denial-of-Service security check violations seen by the Application Firewall.

XML Message Validation (msgvalViolations)

Number of XML Message Validation security check violations seen by the Application Firewall.

Web Services Interoperability (wsIViolations)

Number of Web Services Interoperability (WS-I) security check violations seen by the Application Firewall.

XML SQL Injection (xmlSqlViolations)

Number of XML SQL Injection security check violations seen by the Application Firewall.

XML Cross-Site Scripting (xmlXssViolations)

Number of XML Cross-Site Scripting (XSS) security check violations seen by the Application Firewall.

XML Attachment (xmlAttachmentViolations)

Number of XML Attachment security check violations seen by the Application Firewall.

SOAP Fault Violations (soapflt)

Number of requests returning soap:fault from the backend server

XML Generic Violations (genflt)

Number of requests returning XML generic violation from the backend server

Total Violations (totperpr)

Number of violations seen by the application firewall on per profile basis

HTTP Client Errors (4xx Resp) (4xxResps)

Number of requests returning HTTP 4xx from the backend server

HTTP Server Errors (5xx Resp) (5xxResps)

Number of requests returning HTTP 5xx from the backend server

Example

```
stat appfw profile
```

archive appfw profile

Create archive for the profile.

Synopsys

archive appfw profile <name> <archivename> [-comment <string>]

Arguments

name

Name for the profile. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) characters. Cannot be changed after the profile is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

archivename

Source for tar archive.

comment

Any comments about the purpose of profile, or other useful information about the profile.

restore appfw profile

Restore configuration from archive file

Synopsys

restore appfw profile <archivename>

Arguments

archivename

Source for tar archive.

appfw settings

Sep 22, 2015

The following operations can be performed on "appfw settings":

set | unset | show

set appfw settings

Modifies the global application firewall settings. The global settings apply to all application firewall profiles.

Synopsys

set appfw settings [-defaultProfile <string>] [-undefAction <string>] [-sessionTimeout <positive_integer>] [-learnRateLimit <positive_integer>] [-sessionLifetime <positive_integer>] [-sessionCookieName <string>] [-clientIPLoggingHeader <string>] [-importSizeLimit <positive_integer>] [-signatureAutoUpdate (ON | OFF)] [-signatureUrl <expression>] [-cookiePostEncryptPrefix <string>] [-logMalformedReq (ON | OFF)] [-CEFLogging (ON | OFF)] [-entityDecoding (ON | OFF)] [-useConfigurableSecretKey (ON | OFF)]

Arguments

defaultProfile

Profile to use when a connection does not match any policy. Default setting is APPFW_BYPASS, which sends unmatched connections back to the NetScaler appliance without attempting to filter them further.

Default value: AS_ENGINESETTINGS_DEFAULT_PROF_DEFAULT

undefAction

Profile to use when an application firewall policy evaluates to undefined (UNDEF).

An UNDEF event indicates an internal error condition. The APPFW_BLOCK built-in profile is the default setting. You can specify a different built-in or user-created profile as the UNDEF profile.

Default value: AS_ENGINESETTINGS_UNDEF_PROF_DEFAULT

sessionTimeout

Timeout, in seconds, after which a user session is terminated. Before continuing to use the protected web site, the user must establish a new session by opening a designated start URL.

Default value: AS_ENGINESETTINGS_SESSIONTIMEOUT_DEFAULT

Minimum value: 1

Maximum value: 65535

learnRateLimit

Maximum number of connections per second that the application firewall learning engine examines to generate new relaxations for learning-enabled security checks. The application firewall drops any connections above this limit from the list of connections used by the learning engine.

Default value: AS_ENGINESETTINGS_LEARN_RATE_LIMIT_DEFAULT

Minimum value: 1

Maximum value: 1000

sessionLifetime

Maximum amount of time (in seconds) that the application firewall allows a user session to remain active, regardless of user activity. After this time, the user session is terminated. Before continuing to use the protected web site, the user must establish a new session by opening a designated start URL.

Default value: AS_ENGINESETTINGS_SESSIONLIFETIME_DEFAULT

Maximum value: 2147483647

sessionCookieName

Name of the session cookie that the application firewall uses to track user sessions.

Must begin with a letter or number, and can consist of from 1 to 31 letters, numbers, and the hyphen (-) and underscore (_) symbols.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cookie name" or 'my cookie name').

Default value: NS_S_AS_DEFAULT_COOKIE_NAME

clientIPLoggingHeader

Name of an HTTP header that contains the IP address that the client used to connect to the protected web site or service.

importSizeLimit

Cumulative total maximum number of bytes in web forms imported to a protected web site. If a user attempts to upload files with a total byte count higher than the specified limit, the application firewall blocks the request.

Default value: AS_ENGINESETTINGS_IMPORTSIZELIMIT_DEFAULT

Minimum value: 1

Maximum value: 134217728

signatureAutoUpdate

Flag used to enable/disable auto update signatures

Possible values: ON, OFF

Default value: OFF

signatureUrl

URL to download the mapping file from server

Default value: AS_ENGINESETTINGS_SIGNATURES_UPDATE_URL

cookiePostEncryptPrefix

String that is prepended to all encrypted cookie values.

Default value: NS_S_AS_DEFAULT_CKI_POST_ENCRYPT_PREFIX

logMalformedReq

Log requests that are so malformed that application firewall parsing doesn't occur.

Possible values: ON, OFF

Default value: ON

CEFLogging

Enable CEF format logs.

Possible values: ON, OFF

Default value: OFF

entityDecoding

Transform multibyte (double- or half-width) characters to single width characters.

Possible values: ON, OFF

Default value: OFF

useConfigurableSecretKey

Use configurable secret key in AppFw operations

Possible values: ON, OFF

Default value: OFF

---

unset appfw settings

Use this command to remove appfw settings settings.Refer to the set appfw settings command for meanings of the arguments.

Synopsys

unset appfw settings [-defaultProfile] [-undefAction] [-sessionTimeout] [-learnRateLimit] [-sessionLifetime] [-sessionCookieName] [-clientIPLoggingHeader] [-importSizeLimit] [-signatureAutoUpdate] [-signatureUrl] [-cookiePostEncryptPrefix] [-logMalformedReq] [-CEFLogging] [-entityDecoding] [-useConfigurableSecretKey]

show appfw settings

Displays the current application firewall global settings.

Synopsys

show appfw settings

Arguments

format

level

Outputs

defaultProfile

Profile to use when a connection does not match any policy. Default setting is APPFW_BYPASS, which sends unmatched connections back to the NetScaler appliance without attempting to filter them further.

undefAction

Profile to use when an application firewall policy evaluates to undefined (UNDEF).

An UNDEF event indicates an internal error condition. The APPFW_BLOCK built-in profile is the default setting. You can specify a different built-in or user-created profile as the UNDEF profile.

sessionTimeout

Session timeout (in seconds).

learnRateLimit

Learn messages rate limit value (in messages per second).

sessionLifetime

Session lifetime (in seconds). Zero means no limit.

sessionCookieName

Name of the session cookie that the application firewall uses to track user sessions.

Must begin with a letter or number, and can consist of from 1 to 31 letters, numbers, and the hyphen (-) and underscore (_) symbols.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cookie name" or 'my cookie name').

clientIPLoggingHeader

Name of header that holds downstream IP address for logging purposes.

importSizeLimit

Cumulative total maximum number of bytes in web forms imported to a protected web site. If a user attempts to upload files with a total byte count higher than the specified limit, the application firewall blocks the request.

signatureAutoUpdate

Flag used to enable/disable auto update signatures

signatureUrl

URL to download the mapping file from server

cookiePostEncryptPrefix

String that is prepended to all encrypted cookie values.

logMalformedReq

Log requests that are so malformed that application firewall parsing doesn't occur.

CEFLogging

Enable CEF format logs.

entityDecoding

Transform multibyte (double- or half-width) characters to single width characters.

useConfigurableSecretKey

Use configurable secret key in AppFw operations

appfw signatures

Sep 22, 2015

The following operations can be performed on "appfw signatures":

rm | show | import | update

Removes the specified signature object from the application firewall.

rm appfw signatures <name>

name

Name of the signature object.

rm signatures <name>

Displays the specified signatures object. If no signatures object is specified, displays all signatures objects defined on the NetScaler appliance.

show appfw signatures [<name>]

name

Name of the signature object.

response

show appfw signatures

Imports the specified signatures object to the NetScaler appliance and assigns it the specified name.

import appfw signatures <src> <name> [-xslt <string>] [-comment <string>] [-overwrite] [-merge] [-sha1 <string>]

src

URL (protocol, host, path, and file name) for the location at which to store the imported signatures object.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the signatures object on the NetScaler appliance.

xslt

XSLT file source.

comment

Any comments to preserve information about the signatures object.

overwrite

Overwrite any existing signatures object of the same name.

merge

Merges the existing Signature with new signature rules

sha1

File path for sha1 file to validate signature file

import signatures http://www.example.com/ns/signatures.xml my-signature

---

Updates the specified signatures object from the source.

update appfw signatures <name> [-mergeDefault]

name

Name of the signatures object to update.

mergeDefault

Merges signature file with default signature file.

```
update signatures my-signatures
```

appfw stats

Sep 22, 2015

The following operations can be performed on "appfw stats":

show appfw stats is an alias for stat appfw

show appfw stats - alias for 'stat appfw'

appfw transactionRecords

Sep 22, 2015

The following operations can be performed on "appfw transactionRecords":

Display an application firewall transaction record.

show appfw transactionRecords

httpTransactionId

The http transaction identifier.

packetEngineId

The packet engine identifier.

AppFwSessionId

The session identifier set by the Application Firewall to track the user session.

profileName

Application Firewall profile name.

url

Request URL

clientip

The IP address of client.

destIP

The IP address of destination.

startTime

Conveys time at which request processing started.

endTime

Conveys time at which request processing end.

requestContentLength

The content length of request.

requestYields

The number of times yielded during request processing to send heart beat packets.

requestMaxProcessingTime

The maximum processing time across yields during request processing.

responseContentLength

The content length of response.

responseYields

The number of times yielded during response processing to send heart beat packets.

responseMaxProcessingTime

The maximum processing time across yields during response processing.

flag

Record flags.

devno

count

stateflag

appfw wsdl

Sep 22, 2015

The following operations can be performed on "appfw wsdl":

rm | show | import

Removes the specified imported WSDL file from the application firewall.

rm appfw wsdl <name>

name

Name of the WSDL file to remove.

rm wsdl <name>

Removes the specified imported WSDL file.

show appfw wsdl [<name>]

name

Name of the WSDL file to display.

response

show appfw wsdl

Imports the specified WSDL file to the application firewall.

import appfw wsdl <src> <name> [-comment <string>] [-overwrite]

src

URL (protocol, host, path, and name) of the WSDL file to be imported is stored.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the WSDL on the NetScaler appliance.

comment

Any comments to preserve information about the WSDL.

overwrite

Overwrite any existing WSDL of the same name.

import appfw wsdl http://www.webservicex.net/stockquote.asmx?wsdl stockquote

appfw xmlerrorpage

Sep 22, 2015

The following operations can be performed on "appfw xmlerrorpage":

rm | show | import | update

Removes the object imported by import xmlerrorpage.

rm appfw xmlerrorpage <name>

name

Indicates name of the imported xml error page to be removed.

rm xmlerrorpage <name>

Displays the specified XML error object. If no XML error page object is specified, displays a list of all XML error objects on the NetScaler appliance.

show appfw xmlerrorpage [<name>]

name

Name of the XML error object.

response

show appfw xmlerrorpage

Imports the specified XML error page to the NetScaler appliance and assigns it the specified name.

import appfw xmlerrorpage <src> <name> [-comment <string>] [-overwrite]

src

URL (protocol, host, path, and name) for the location at which to store the imported XML error object.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the XML error object on the NetScaler appliance.

comment

Any comments to preserve information about the XML error object.

overwrite

Overwrite any existing XML error object of the same name.

import xmlerrorpage http://www.example.com/errorpage.xml my-xml-error-page

---

Updates the specified XML error object from the source.

update appfw xmlerrorpage <name>

name

Name of the XML error object.

update xmlerrorpage my-xml-error-page

---

appfw xmlschema

Sep 22, 2015

The following operations can be performed on "appfw xmlschema":

rm | show | import

Removes the specified XML Schema object from the application firewall.

rm appfw xmlschema <name>

name

Name of the XML Schema object to remove.

rm xmlschema <name>

Displays the specified XML Schema object. If no object is specified, displays all XML Schema objects on the NetScaler appliance.

show appfw xmlschema [<name>]

name

Name of the XML Schema object to display.

response

show appfw xmlschema

Imports the specified XML Schema to the NetScaler appliance and assigns it the specified name.

import appfw xmlschema <src> <name> [-comment <string>] [-overwrite]

src

URL (protocol, host, path, and file name) for the location at which to store the imported XML Schema.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the XML Schema object on the NetScaler appliance.

comment

Any comments to preserve information about the XML Schema object.

overwrite

Overwrite any existing XML Schema object of the same name.

import xmlschema http://schemas.xmlsoap.org/soap/envelope/ soap

AppQoE Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- appqoe
- appqoe CustomResp
- appqoe action
- appqoe parameter
- appqoe policy
- appqoe stats

appqoe

Sep 22, 2015

The following operations can be performed on "appqoe":

Displays statistics of feature AppQoE.

stat appqoe [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

In-Memory responses sent (TotInMemRsp)

Total in-memory responses sent from NS

Faulty cookies received (TotFaultyCookies)

Total faulty cookies received

Valid cookies received (TotValidCookies)

Total valid cookies received

High priority requests served (TotHighPriReq)

Total Requests served from higher priority queue

Medium priority requests served (TotMediumPriReq)

Total Requests served from medium priority queue

Low priority requests served (TotLowPriReq)

Total Requests served from low priority queue

Lowest(Surge) priority requests served (TotLowestPriReq)

Total Requests served from surge priority queue

Alt. server substitution failed (TotAltSvrSubFailed)

Total number of times alternate server substitution failed

HDOS condition triggered (TotDoSTrig)

Total number of times HDOS condition triggered

Valid DOSQ cookies received (TotDOSQValidCookies)

Total DOSQ valid cookies received

Valid DOSH cookies received (TotDOSHValidCookies)

Total DOSH valid cookies received

Valid SID cookies received (TotSIDValidCookies)

Total SID valid cookies received

Valid ONH cookies received (TotONHValidCookies)

Total ONH valid cookies received

Valid PRIQ cookies received (TotPRIQValidCookies)

Total PRIQ valid cookies received

Faulty DOSQ cookies received (TotDOSQFaultyCookies)

Total DOSQ faulty cookies received

Faulty DOSH cookies received (TotDOSHFaultyCookies)

Total DOSH faulty cookies received

Faulty SID cookies received (TotSIDFaultyCookies)

Total SID faulty cookies received

Faulty ONH cookies received (TotONHFaultyCookies)

Total ONH faulty cookies received

Faulty PRIQ cookies received (TotPRIQFaultyCookies)

Total PRIQ faulty cookies received

Requests for valid embedded links (TotPRIEmbedLinks)

Total requests for valid embedded links

Valid SIDQ req. within session (TotSessReq)

Total valid SIDQ requests within session

Requests for alternate contents (TotAltCntReq)

Total requests for alternate contents

In-Memory GET responses sent (TotGETInMemRsp)

Total in-memory GET responses sent from NS

In-Memory POST responses sent (TotPOSTInMemRsp)

Total in-memory POST responses sent from NS

In-Memory response bytes sent (TotInMemRspbytes)

Total in-memory response bytes sent from NS

appqoe CustomResp

Sep 22, 2015

The following operations can be performed on "appqoe CustomResp":

import | rm | show | update

Downloads the input HTML Page to NetScaler Box with the given object name

import appqoe CustomResp [<src>] <name>

src

name

Indicates name of the custom response HTML page to import/update.

import appqoe CustomResp http://10.102.34.25/index.html appqoe_resp

---

Removes the imported HTML object.

rm appqoe CustomResp <name>

name

Indicates name of the custom response HTML page to import/update.

rm appqoe CustomResp appqoe_resp

Displays lists all HTML page objects on the NetScaler appliance.

show appqoe CustomResp

summary

fullValues

name

Indicates name of the custom response HTML page to import/update.

src

devno

count

stateflag

show appqoe CustomResp

Update the imported HTML object

update appqoe CustomResp <name>

name

Indicates name of the custom response HTML page to import/update.

update appqoe CustomResp appqoe_resp

---

appqoe action

Sep 22, 2015

The following operations can be performed on "appqoe action":

add | rm | set | unset | show

Add a new AppQoE action for triggering

add appqoe action <name> [-priority <priority>] [-respondWith (ACS | NS) [<CustomFile>] [-altContentSvcName <string>] [-altContentPath <string>] [-maxConn <positive_integer>] [-delay <usecs>]] [-polqDepth <positive_integer>] [-priqDepth <positive_integer>] [-dosTrigExpression <expression>] [-dosAction (SimpleResponse | HICResponse)]

name

Name for the AppQoE action. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), and colon (:) characters. This is a mandatory argument

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. If priority is not configured then Lowest priority will be used to queue the request.

Possible values: HIGH, MEDIUM, LOW, LOWEST

respondWith

Responder action to be taken when the threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service

Threshold : maxConn or delay

NS - Serve from the NetScaler appliance (built-in response)

Threshold : maxConn or delay

Possible values: ACS, NS

CustomFile

name of the HTML page object to use as the response

altContentSvcName

Name of the alternative content service to be used in the ACS

altContentPath

Path to the alternative content service to be used in the ACS

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests queued for the policy binding this action is attached to) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

priqDepth

Queue depth threshold value per priorirty level. If the queue size (number of requests in the queue of that particular priorirty) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

maxConn

Maximum number of concurrent connections that can be open for requests that matches with rule.

Minimum value: 1

Maximum value: 4294967294

delay

Delay threshold, in microseconds, for requests that match the policy's rule. If the delay statistics gathered for the matching request exceed the specified delay, configured action triggered for that request, if there is no action then requests are dropped to the lowest priority level

Minimum value: 1

Maximum value: 599999999

dosTrigExpression

Optional expression to add second level check to trigger DoS actions. Specifically used for Analytics based DoS response generation

dosAction

DoS Action to take when vserver will be considered under DoS attack and corresponding rule matches. Mandatory if AppQoE actions are to be used for DoS attack prevention.

Possible values: SimpleResponse, HICResponse

Removes the specified AppQoE action.

rm appqoe action <name>

name

Name of the action to be removed.

Set the argument of specified AppQoE action.

set appqoe action <name> [-priority <priority>] [-altContentSvcName <string>] [-altContentPath <string>] [-polqDepth <positive_integer>] [-priqDepth <positive_integer>] [-maxConn <positive_integer>] [-delay <usecs>] [-dosTrigExpression <expression>] [-dosAction (SimpleResponse | HICResponse)]

name

Name for the AppQoE action. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), and colon (:) characters. This is a mandatory argument

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. If priority is not configured then Lowest priority will be used to queue the request.

Possible values: HIGH, MEDIUM, LOW, LOWEST

altContentSvcName

Name of the alternative content service to be used in the ACS

altContentPath

Path to the alternative content service to be used in the ACS

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests queued for the policy binding this action is attached to) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

priqDepth

Queue depth threshold value per priorirty level. If the queue size (number of requests in the queue of that particular priorirty) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

maxConn

Maximum number of concurrent connections that can be open for requests that matches with rule.

Minimum value: 1

Maximum value: 4294967294

delay

Delay threshold, in microseconds, for requests that match the policy's rule. If the delay statistics gathered for the matching request exceed the specified delay, configured action triggered for that request, if there is no action then requests are dropped to the lowest priority level

Minimum value: 1

Maximum value: 599999999

dosTrigExpression

Optional expression to add second level check to trigger DoS actions. Specifically used for Analytics based DoS response generation

dosAction

DoS Action to take when vserver will be considered under DoS attack and corresponding rule matches. Mandatory if AppQoE actions are to be used for DoS attack prevention.

Possible values: SimpleResponse, HICResponse

Use this command to remove appqoe action settings.Refer to the set appqoe action command for meanings of the arguments.

unset appqoe action <name> [-priority] [-altContentSvcName] [-altContentPath] [-polqDepth] [-priqDepth] [-maxConn] [-delay] [-dosAction]

Display configured AppQoE action(s).

show appqoe action [<name>]

name

Name for the AppQoE action. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), and colon (:) characters. This is a mandatory argument

summary

fullValues

format

level

stateflag

hits

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. If priority is not configured then Lowest priority will be used to queue the request.

respondWith

Responder action to be taken when the threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service

Threshold : maxConn or delay

NS - Serve from the NetScaler appliance (built-in response)

Threshold : maxConn or delay

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests queued for the policy binding this action is attached to)

increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

priqDepth

Queue depth threshold value per priorirty level. If the queue size (number of requests in the queue of that particular priorirty) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

altContentSvcName

Name of the alternative content service to be used in the ACS

altContentPath

Path to the alternative content service to be used in the ACS

maxConn

Maximum number of concurrent connections that can be open for requests that matches with rule.

delay

Delay threshold, in microseconds, for requests that match the policy's rule. If the delay statistics gathered for the matching request exceed the specified delay, configured action triggered for that request, if there is no action then requests are dropped to the lowest priority level

CustomFile

name of the HTML page object to use as the response

dosTrigExpression

Optional expression to add second level check to trigger DoS actions. Specifically used for Analytics based DoS response generation

dosAction

DoS Action to take when vserver will be considered under DoS attack and corresponding rule matches. Mandatory if AppQoE actions are to be used for DoS attack prevention.

devno

count

appqoe parameter

Sep 22, 2015

The following operations can be performed on "appqoe parameter":

set | unset | show

Sets the parameters for displaying appqoe information.

set appqoe parameter [-sessionLife <secs>] [-avgwaitingclient <positive_integer>] [-MaxAltRespBandWidth <positive_integer>] [-dosAttackThresh <positive_integer>]

sessionLife

Time, in seconds, between the first time and the next time the AppQoE alternative content window is displayed. The alternative content window is displayed only once during a session for the same browser accessing a configured URL, so this parameter determines the length of a session.

Default value: 300

Minimum value: 1

Maximum value: 4294967294

avgwaitingclient

average number of client connections, that can sit in service waiting queue

Default value: 1000000

Maximum value: 4294967294

MaxAltRespBandWidth

maximum bandwidth which will determine whether to send alternate content response

Default value: 100

Minimum value: 1

Maximum value: 4294967294

dosAttackThresh

average number of client connection that can queue up on vserver level without triggering DoS mitigation module

Default value: 2000

Maximum value: 4294967294

set appqoe parameter -sessionlife 200 -avgwaitingclient 10

Use this command to remove appqoe parameter settings.Refer to the set appqoe parameter command for meanings of the arguments.

unset appqoe parameter [-sessionLife] [-avgwaitingclient] [-MaxAltRespBandWidth] [-dosAttackThresh]

Displays the values of the session life and filename parameters

show appqoe parameter

format

level

sessionLife

appqoe session life (in seconds)

avgwaitingclient

average number of client connections, that can sit in service waiting queue

MaxAltRespBandWidth

maximum bandwidth which will determine whether to send alternate content response

dosAttackThresh

average number of client connection that can queue up on vserver level without triggering DoS mitigation module

show appqos parameter

appqoe policy

Sep 22, 2015

The following operations can be performed on "appqoe policy":

add | rm | set | show | stat

Add a new AppQoE policy for binding rule with action

add appqoe policy <name> -rule <expression> -action <string>

name

rule

Expression or name of a named expression, against which the request is evaluated. The policy is applied if the rule evaluates to true.

action

Configured AppQoE action to trigger

Remove an AppQoE policy.

rm appqoe policy <name>

name

Name of the AppQoE policy to be removed.

---

set appqoe policy <name> [-rule <expression>] [-action <string>]

name

rule

Expression or name of a named expression, against which the request is evaluated. The policy is applied if the rule evaluates to true.

action

Configured AppQoE action to trigger

Display all the configured AppQoE policies.

show appqoe policy [<name>]

name

summary

fullValues

format

level

stateflag

rule

Expression or name of a named expression, against which the request is evaluated. The policy is applied if the rule evaluates to true.

action

Configured AppQoE action to trigger

hits

Number of hits.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

bindPriority

Specifies the binding of the policy. use only in display

boundTo

The name of the entity to which the policy is bound.

activePolicy

devno

count

Displays collected brief statistics for all AppQoE policies, or detailed statistics for only the specified policy.

stat appqoe policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

policyName

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Server TTFB (SvrTTFB)

Server Time-To-First-Byte in milliseconds calculated for this AppQoE policy.

Server TTLB (SvrTTLB)

Server Time-To-Last-Byte in milliseconds

calculated for this AppQoE policy.

Client TTLB (CltTTLB)

Client Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

Avg. Server TTFB (SvrTTFB)

Average Server Time-To-First-Byte in milliseconds calculated for this AppQoE policy.

Avg. Server TTLB (SvrTTLB)

Average Server Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

Avg. Client TTLB (CltTTLB)

Average Client Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

ThroughPut(KBps) (ThroughPut)

Throughput in KBps calculated on this AppQoE policy

Server TCP connections (TotSvr)

Total number of server connections that were established through this AppQoE Policy

Client TCP connections requested (TotClt)

Total number of client connections that were requested through this AppQoE Policy

Requests received (TotReq)

Total number of requests that were requested through this AppQoE policy

Requests bytes received (TotReqBytes)

Total number of requests bytes that were requested through this AppQoE

policy

Responses received (TotRsp)

Total number of responses received by this AppQoE policy

Response bytes received (TotRspBytes)

Total number of response bytes received by this AppQoE policy

Alternate responses sent (TotJSsent)

Total number of in-memory responses sent instead of expected responses through this AppQoE policy

Alternate responses bytes sent (TotJSBytessent)

Total bytes of in-memory responses sent through this AppQoE policy

Policy hits (Hits)

Number of hits on the policy

Client HTTP transactions

Total number of client transactions processed by this AppQoE policy.

Svr HTTP transactions

Total number of server transactions processed by this AppQoE policy.

stat appqos policy

appqoe stats

Sep 22, 2015

The following operations can be performed on "appqoe stats":

show appqoe stats is an alias for stat appqoe Displays global AppQoE statistics.

show appqoe stats - alias for 'stat appqoe'

Audit Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- audit
- audit messageaction
- audit messages
- audit nslogAction
- audit nslogParams
- audit nslogPolicy
- audit stats
- audit syslogAction
- audit syslogParams
- audit syslogPolicy

audit

Sep 22, 2015

The following operations can be performed on "audit":

Display the audit statistics

stat audit [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Audit logs sent to syslog server(s) (LogSnd)

Syslog messages sent to the syslog server(s).

Audit log messages generated (LogGen)

Syslog messages about to be sent to the syslog server.

NAT allocation failed (Ernatpcb)

NAT allocation failed.

Nsb allocation failed (Ernsb)

Nsb allocation failed.

Memory allocation failed (Ermem)

Failures in allocation of Access Gateway context structure. When an Access Gateway session is established, the NetScaler creates an internal context structure , which identifies the user and the IP address from which the user has logged in.

Port allocation failed (Erport)

Number of times the NetScaler failed to allocate a port when sending a syslog message to the syslog server(s).

NAT lookup failed (Hshmiss)

NAT lookup failed.

Context not found (Ctxntfnd)

Failures in finding the context structure for an Access Gateway session during attempts to send session-specific audit messages.

During an Access Gateway session, audit messages related to the session are queued up in the auditlog buffer for transmission to the audit log server(s). If the session is killed before the messages are sent, the context structure allocated at session creation is removed. This structure is needed for sending the queued auditlog messages. If it is not found, this counter is incremented.

Nsb chain allocation failed (Ernsbchn)

Nsb Chain allocation failed.

Client connect failed (Erclconn)

Failures in establishment of a connection between the NetScaler and the auditserver tool (the Netscaler's custom logging tool).

MP buffer flush command count (flcmdcnt)

Auditlog buffer flushes. In a multiprocessor NetScaler, both the main processor and the co-processor can generate auditlog messages and fill up the auditlog buffers. But only the primary processor can free up the buffers by sending auditlog messages to the auditlog server(s). The number of auditlog buffers is fixed. If the co-processor detects that all the auditlog buffers are full, it issues a flush command to the main processor.

audit messageaction

Sep 22, 2015

The following operations can be performed on "audit messageaction":

add | rm | set | unset | show

Adds an audit message action. The action specifies whether to log the message, and to which log.

add audit messageaction <name> <logLevel> <stringBuilderExpr> [-logtoNewnslog (YES | NO)] [-bypassSafetyCheck (YES | NO)]

name

Name of the audit message action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the message action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my message action? or ?my message action?).

logLevel

Audit log level, which specifies the severity level of the log message being generated..

The following loglevels are valid:

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

Possible values: EMERGENCY, ALERT, CRITICAL, ERROR, WARNING, NOTICE, INFORMATIONAL, DEBUG

stringBuilderExpr

Default-syntax expression that defines the format and content of the log message.

logtoNewnslog

Send the message to the new nslog.

Possible values: YES, NO

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions.

Possible values: YES, NO

Default value: NO

Removes the specified audit message action and associated configuration.

rm audit messageaction <name>

name

Name of the audit message action to remove.

Modifies the specified parameters of an existing audit message action.

set audit messageaction <name> [-logLevel <logLevel>] [-stringBuilderExpr <string>] [-logtoNewnslog (YES | NO)] [-bypassSafetyCheck (YES | NO)]

name

Name of the audit message action to modify.

logLevel

Audit log level, which specifies the severity level of the log message being generated.

The following loglevels are valid:

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

Possible values: EMERGENCY, ALERT, CRITICAL, ERROR, WARNING, NOTICE, INFORMATIONAL, DEBUG

stringBuilderExpr

Default-syntax expression that defines the format and content of the log message.

logtoNewnslog

Send the message to the new nslog.

Possible values: YES, NO

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions.

Possible values: YES, NO

Default value: NO

Use this command to remove audit messageaction settings.Refer to the set audit messageaction command for meanings of the arguments.

```
unset audit messageaction <name> [-logtoNewnslog] [-bypassSafetyCheck]
```

Displays the current configuration of the specified audit message action. If no audit message action is specified, displays a list of all audit message actions currently configured on the NetScaler appliance.

```
show audit messageaction [<name>]
```

name

Name of the audit message action.

summary

fullValues

format

level

logLevel

stringBuilderExpr

Default-syntax expression that defines the format and content of the log message.

logtoNewnslog

Send the message to the new nslog.

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions.

stateflag

hits

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

devno

count

audit messages

Sep 22, 2015

The following operations can be performed on "audit messages":

Displays the most recent audit log messages.

show audit messages [-logLevel <logLevel> ...] [-numOfMesgs <positive_integer>]

logLevel

Audit log level filter, which specifies the types of events to display.

The following loglevels are valid:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

numOfMesgs

Number of log messages to be displayed.

Default value: 20

Minimum value: 1

Maximum value: 256

summary

fullValues

value

The Audit message

devno

count

stateflag

audit nslogAction

Sep 22, 2015

The following operations can be performed on "audit nslogAction":

add | rm | set | unset | show

Adds an nslog action. The action contains a reference to an nslog server and specifies which information to log and how to log that information.

add audit nslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> ... [-dateFormat <dateFormat>] [-logFacility <logFacility>] [-tcp (NONE | ALL)] [-acl (ENABLED | DISABLED)] [-timeZone (GMT_TIME | LOCAL_TIME)] [-userDefinedAuditlog (YES | NO)] [-appflowExport (ENABLED | DISABLED)]

name

Name of the nslog action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the nslog action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my nslog action? or ?my nslog action).

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME. Coordinated Universal Time.

* LOCAL_TIME. The server?s timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

Removes the specified nslog action and associated configuration. Note: An nslog action cannot be removed if it is bound to an nslog policy.

rm audit nslogAction <name>

name

Name of the nslog action to remove.

Modifies the specified settings of an existing nslog action.

set audit nslogAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-logLevel <logLevel> ...] [-dateFormat <dateFormat>] [-logFacility <logFacility>] [-tcp (NONE | ALL)] [-acl (ENABLED | DISABLED)] [-timeZone (GMT_TIME | LOCAL_TIME)] [-userDefinedAuditlog (YES | NO)] [-appflowExport (ENABLED | DISABLED)]

name

Name of the nslog action to be modified.

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME. Coordinated Universal Time.

* LOCAL_TIME. The server?s timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

Removes the settings of an existing nslog action. Attributes for which a default value is available revert to their default values. See the set audit nslogAction command for descriptions of the parameters..Refer to the set audit nslogAction command for meanings of the arguments.

unset audit nslogAction <name> [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport]

Displays the current configuration of the specified nslog action. If no nslog action is specified, displays a list of all nslog actions currently configured on the NetScaler appliance.

show audit nslogAction [<name>]

name

Name of the nslog action.

summary

fullValues

format

level

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

logLevel

Audit log level, which specifies the types of events to log.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Log TCP messages.

acl

Log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME. Coordinated Universal Time.

* LOCAL_TIME. The server?s timezone setting.

stateflag

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

audit nslogParams

Sep 22, 2015

The following operations can be performed on "audit nslogParams":

set | unset | show

Modifies the specified nslog parameters. Changes the IP address, the port, or the logging parameters for logs sent to nslog.

set audit nslogParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-dateFormat <dateFormat>] [-logLevel <logLevel> ...] [-logFacility <logFacility>] [-tcp (NONE | ALL)] [-acl (ENABLED | DISABLED)] [-timeZone (GMT_TIME | LOCAL_TIME)] [-userDefinedAuditlog (YES | NO)] [-appflowExport (ENABLED | DISABLED)]

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

Minimum value: 1

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logLevel

Types of information to be logged.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Configure auditing to log TCP messages.

Possible values: NONE, ALL

acl

Configure auditing to log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME - Coordinated Universal Time.

* LOCAL_TIME - Use the server?s timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

Removes the existing nslog parameter settings. Attributes for which a default value is available revert to their default values. See the set audit nslogParams command for a description of the parameters..Refer to the set audit nslogParams command for meanings of the arguments.

unset audit nslogParams [-serverIP] [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport]

Displays the current nslog parameter settings.

show audit nslogParams

format

level

name

Name of the nslog param.NOTE: This attribute is deprecated.This argument is deprecated since for syslog and nslogparms there is no name.

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

logLevel

The audit log level.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Configure auditing to log TCP messages.

acl

Configure auditing to log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME - Coordinated Universal Time.

* LOCAL_TIME - Use the server?s timezone setting.

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

audit nslogPolicy

Sep 22, 2015

The following operations can be performed on "audit nslogPolicy":

add | rm | set | show

Adds a policy that defines which messages to log to the specified nslog server.

add audit nslogPolicy <name> <rule> <action>

name

Name for the policy.

Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the nslog policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my nslog policy? or ?my nslog policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the nslog server.

action

Nslog server action that is performed when this policy matches.

NOTE: An nslog server action must be associated with an nslog audit policy.

Removes the specified nslog policy and associated configuration.

rm audit nslogPolicy <name>

name

Name of the nslog policy to remove.

Modifies the specified parametrers of an existing nslog policy.

set audit nslogPolicy <name> [-rule <expression>] [-action <string>]

name

Name of the nslog policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the nslog server.

action

Nslog server action that is performed when this policy matches.

NOTE: An nslog server action must be associated with an nslog audit policy.

Displays the current configuration of the specified nslog policy. If no nslog policy is specified, displays a list of all nslog policies currently configured on the NetScaler appliance.

show audit nslogPolicy [<name>]

name

Name of the policy.

summary

fullValues

format

level

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the nslog server.

action

Nslog server action that is performed when this policy matches.

NOTE: An nslog server action must be associated with an nslog audit policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

audit stats

Sep 22, 2015

The following operations can be performed on "audit stats":

show audit stats

show audit stats is an alias for stat audit

Synopsys

show audit stats - alias for 'stat audit'

audit syslogAction

Sep 22, 2015

The following operations can be performed on "audit syslogAction":

add | rm | set | unset | show

add audit syslogAction

Adds a syslog action. The action contains a reference to a syslog server, and specifies which information to log and how to log that information.

Synopsys

add audit syslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> ... [-dateFormat <dateFormat>] [-logFacility <logFacility>] [-tcp (NONE | ALL)] [-acl (ENABLED | DISABLED)] [-timeZone (GMT_TIME | LOCAL_TIME)] [-userDefinedAuditlog (YES | NO)] [-appflowExport (ENABLED | DISABLED)]

Arguments

name

Name of the syslog action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the syslog action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my syslog action? or ?my syslog action).

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available values function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY. -U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME. Coordinated Universal time.

* LOCAL_TIME. Use the server?s timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

---

rm audit syslogAction

Removes the specified syslog action and associated configuration. Note: A syslog action cannot be removed if it is bound to a syslog policy.

Synopsys

rm audit syslogAction <name>

Arguments

name

Name of the syslog action to remove.

---

set audit syslogAction

Modifies the specified parameters of an existing syslog action.

Synopsys

set audit syslogAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-logLevel <logLevel> ...] [-dateFormat <dateFormat>] [-logFacility <logFacility>] [-tcp (NONE | ALL)] [-acl (ENABLED | DISABLED)] [-timeZone (GMT_TIME | LOCAL_TIME)] [-userDefinedAuditlog (YES | NO)] [-appflowExport (ENABLED | DISABLED)]

Arguments

name

Name of the syslog action to be modified.

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available values function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY. -U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME. Coordinated Universal time.

* LOCAL_TIME. Use the server?s timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

---

unset audit syslogAction

Removes the settings of an existing syslog action. Attributes for which a default value is available revert to their default values. See the set audit syslogAction command for a description of the parameters..Refer to the set audit syslogAction command for meanings of the arguments.

Synopsys

unset audit syslogAction <name> [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport] [-serverIP]

show audit syslogAction

Displays the current configuration of the specified syslog action. If no syslog action is specified, displays a list of all syslog actions currently configured on the NetScaler appliance.

Synopsys

show audit syslogAction [<name>]

Arguments

name

Name of the syslog action.

summary

fullValues

format

level

Outputs

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

logLevel

Audit log level, which specifies the types of events to log.

Available values function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY. -U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Log TCP messages.

acl

Log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME. Coordinated Universal time.

* LOCAL_TIME. Use the server?s timezone setting.

stateflag

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

appflowExport

Disable export of log messages to AppFlow collectors.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

audit syslogParams

Sep 22, 2015

The following operations can be performed on "audit syslogParams":

set | unset | show

set audit syslogParams

Modifies the syslog parameters. Changes the IP, the port, or the logging parameters for logs sent to syslog.

Synopsys

set audit syslogParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-dateFormat <dateFormat>] [-logLevel <logLevel> ...] [-logFacility <logFacility>] [-tcp (NONE | ALL)] [-acl (ENABLED | DISABLED)] [-timeZone (GMT_TIME | LOCAL_TIME)] [-userDefinedAuditlog (YES | NO)] [-appflowExport (ENABLED | DISABLED)]

Arguments

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

Minimum value: 1

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY. European style -date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logLevel

Types of information to be logged.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME - Coordinated Universal Time.

* LOCAL_TIME Use the server?s timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes audit to ignore all user-configured message actions. Setting this parameter to YES causes audit to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

---

unset audit syslogParams

Removes the existing syslog parameter settings. Attributes for which a default value is available revert to their default values. See the set audit syslogParams command for descriptions of the parameters..Refer to the set audit syslogParams command for meanings of the arguments.

Synopsys

unset audit syslogParams [-serverIP] [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport]

---

show audit syslogParams

Displays the current syslog parameter settings.

Synopsys

show audit syslogParams

Arguments

format

level

Outputs

name

Name.NOTE: This attribute is deprecated.This argument is deprecated since for syslog and nslogparms there is no name.

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY. European style -date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

logLevel

Types of information to be logged.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Log TCP messages.

acl

Log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME - Coordinated Universal Time.

* LOCAL_TIME Use the server?s timezone setting.

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes audit to ignore all user-configured message actions. Setting this parameter to YES causes audit to log user-configured message actions that meet the other logging criteria.

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

audit syslogPolicy

Sep 22, 2015

The following operations can be performed on "audit syslogPolicy":

add | rm | set | show

add audit syslogPolicy

Adds a policy that defines which messages to log to the specified syslog server.

Synopsys

add audit syslogPolicy <name> <rule> <action>

Arguments

name

Name for the policy.

Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the syslog policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my syslog policy? or ?my syslog policy).

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the syslog server.

action

Syslog server action to perform when this policy matches traffic.

NOTE: A syslog server action must be associated with a syslog audit policy.

---

rm audit syslogPolicy

Removes the specified syslog policy and associated configuration.

Synopsys

rm audit syslogPolicy <name>

Arguments

name

Name of the syslog policy to remove.

---

set audit syslogPolicy

Configures an existing syslog policy.

Synopsys

set audit syslogPolicy <name> [-rule <expression>] [-action <string>]

Arguments

name

Name of the syslog policy to be configured.

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the syslog server.

action

Syslog server action to perform when this policy matches traffic.

NOTE: A syslog server action must be associated with a syslog audit policy.

---

show audit syslogPolicy

Displays the current configuration of the specified syslog policy. If no syslog policy is specified, displays a list of all syslog policies currently configured on the NetScaler appliance.

Synopsys

show audit syslogPolicy [<name>]

Arguments

name

Name of the policy.

summary

fullValues

format

level

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the syslog server.

action

Syslog server action to perform when this policy matches traffic.

NOTE: A syslog server action must be associated with a syslog audit policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

Authentication Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- authentication authnProfile
- authentication certAction
- authentication certPolicy
- authentication ldapAction
- authentication ldapPolicy
- authentication localPolicy
- authentication negotiateAction
- authentication negotiatePolicy
- authentication radiusAction
- authentication radiusPolicy
- authentication samlAction
- authentication samlPolicy
- authentication tacacsAction
- authentication tacacsPolicy
- authentication vserver

authentication authnProfile

Sep 22, 2015

The following operations can be performed on "authentication authnProfile":

add | rm | set | unset | show

add authentication authnProfile

Creates an authentication profile to hold all authentication related configuration for TM vserver.

Synopsys

add authentication authnProfile <name> {-authnVsName <string>} {-AuthenticationHost <string>} {-AuthenticationDomain <string>} [-AuthenticationLevel <positive_integer>]

Arguments

name

Name for the authentication profile.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the RADIUS action is added.

authnVsName

Name of the authentication vserver at which authentication should be done.

Maximum value: 128

AuthenticationHost

Hostname of the authentication vserver.

Maximum value: 256

AuthenticationDomain

Domain for which TM cookie must to be set. If unspecified, cookie will be set for FQDN.

Maximum value: 256

AuthenticationLevel

---

rm authentication authnProfile

Removes an authentication profile. A profile cannot be removed as long as it is set to a vserver.

Synopsys

rm authentication authnProfile <name>

Arguments

name

Name of the authentication profile to be removed.

---

set authentication authnProfile

Configures an authentication profile.

Synopsys

set authentication authnProfile <name> [-authnVsName <string>] [-AuthenticationHost <string>] [-AuthenticationDomain <string>] [-AuthenticationLevel <positive_integer>]

Arguments

name

Name of the authentication profile.

authnVsName

Name of the authentication vserver at which authentication should be done.

Maximum value: 128

AuthenticationHost

Hostname of the authentication vserver.

Maximum value: 256

AuthenticationDomain

Domain for which TM cookie must to be set. If unspecified, cookie will be set for FQDN.

Maximum value: 256

AuthenticationLevel

---

unset authentication authnProfile

Use this command to remove authentication authnProfile settings.Refer to the set authentication authnProfile command for meanings of the arguments.

Synopsys

unset authentication authnProfile <name> [-AuthenticationDomain] [-AuthenticationLevel]

show authentication authnProfile

Displays the current configuration for the authentication profile specified

Synopsys

show authentication authnProfile [<name>]

Arguments

name

Name of the authentication profile.

summary

fullValues

format

level

Outputs

authnVsName

Name of the authentication vserver at which authentication should be done.

AuthenticationHost

Hostname of the authentication vserver.

AuthenticationDomain

Domain for which TM cookie must to be set. If unspecified, cookie will be set for FQDN.

AuthenticationLevel

devno

count

stateflag

authentication certAction

Sep 22, 2015

The following operations can be performed on "authentication certAction":

add | rm | set | unset | show

add authentication certAction

Adds an action (profile) for a client certificate (cert) authentication server. The profile contains all configuration data necessary to communicate with that client cert authentication server.

Synopsys

add authentication certAction <name> [-twoFactor (ON | OFF)] [-userNameField <string>] [-groupNameField <string>] [-defaultAuthenticationGroup <string>]

Arguments

name

Name for the client cert authentication server profile (action).

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after certifcate action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

twoFactor

Enables or disables two-factor authentication.

Two factor authentication is client cert authentication followed by password authentication.

Possible values: ON, OFF

Default value: OFF

userNameField

Client-cert field from which the username is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>.

groupNameField

Client-cert field from which the group is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

```
add authentication certaction -twoFactor ON -userNameField "Subject:CN" -groupNameField "Subject:OU"
```

---

rm authentication certAction

Removes an existing client cert authentication server profile (action).

Synopsys

```
rm authentication certAction <name>
```

Arguments

name

Name of the profile to be removed.

---

set authentication certAction

Configures a client cert authentication server profile (action).

Synopsys

```
set authentication certAction <name> [-twoFactor ( ON | OFF )] [-userNameField <string>] [-groupNameField <string>] [-defaultAuthenticationGroup <string>]
```

Arguments

name

Name of the client cert server profile.

twoFactor

Enables or disables two-factor authentication.

Two factor authentication is client cert authentication followed by password authentication.

Possible values: ON, OFF

Default value: OFF

userNameField

Client-cert field from which the username is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>.

groupNameField

Client-cert field from which the group is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

```
set authentication certaction -twoFactor ON -userNameField "Subject:CN" -groupNameField "Subject:OU"
```

---

unset authentication certAction

Use this command to remove authentication certAction settings.Refer to the set authentication certAction command for meanings of the arguments.

Synopsys

```
unset authentication certAction <name> [-twoFactor] [-userNameField] [-groupNameField] [-defaultAuthenticationGroup]
```

---

show authentication certAction

Displays the current configuration settings for the specified client cert authentication server profile (action).

Synopsys

```
show authentication certAction [<name>]
```

Arguments

name

Name of the client cert server profile (action).

summary

fullValues

format

level

Outputs

twoFactor

The state of two factor authentication.

userNameField

The field in the certificate from which the username will be extracted.

groupNameField

The field in the certificate from which the group will be extracted.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

stateflag

devno

count

authentication certPolicy

Sep 22, 2015

The following operations can be performed on "authentication certPolicy":

add | rm | set | unset | show

add authentication certPolicy

Adds a client certificate (cert) authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified client cert authentication server.

Synopsys

add authentication certPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the client certificate authentication policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after cert authentication policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the authentication server.

reqAction

Name of the client cert authentication action to be performed if the policy matches.

---

rm authentication certPolicy

Removes a client cert authentication policy.

Synopsys

rm authentication certPolicy <name>

Arguments

name

Name of the client cert policy to remove.

---

set authentication certPolicy

Configures the specified client cert authentication policy.

Synopsys

set authentication certPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the client cert policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the authentication server.

reqAction

Name of the client cert authentication action to be performed if the policy matches.

---

unset authentication certPolicy

Use this command to remove authentication certPolicy settings.Refer to the set authentication certPolicy command for meanings of the arguments.

Synopsys

unset authentication certPolicy <name> [-rule] [-reqAction]

---

show authentication certPolicy

Displays the current settings for the specified client cert authentication policy. If no policy name is provided, displays a list of all client cert authentication policies currently configured on the NetScaler appliance.

Synopsys

show authentication certPolicy [<name>]

Arguments

name

Name of the client cert authentication policy.

summary

fullValues

format

level

Outputs

rule

The rule associated with the policy.

reqAction

The cert action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication ldapAction

Sep 22, 2015

The following operations can be performed on "authentication ldapAction":

add | rm | set | unset | show

add authentication ldapAction

Creates an action (profile) for an LDAP server. This profile contains all configuration data needed to communicate with that LDAP server.

Synopsys

add authentication ldapAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-ldapLoginName <string>] [-searchFilter <string>] [-groupAttrName <string>] [-subAttributeName <string>] [-secType <secType>] [-svrType (AD | NDS)] [-ssoNameAttribute <string>] [-authentication (ENABLED | DISABLED)] [-requireUser (YES | NO)] [-passwdChange (ENABLED | DISABLED)] [-nestedGroupExtraction (ON | OFF) [-maxNestingLevel <positive_integer>] [-groupSearchSubAttribute <string>] [-groupSearchFilter <string>]] [-followReferrals (ON | OFF) [-maxLDAPReferrals <positive_integer>]] [-validateServerCert (YES | NO)] [-ldapHostname <string>] [-groupNameIdentifier <string>] [-groupSearchAttribute <string>] [-defaultAuthenticationGroup <string>]

Arguments

name

Name for the new LDAP action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the LDAP action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

serverIP

IP address assigned to the LDAP server.

serverPort

Port on which the LDAP server accepts connections.

Default value: 389

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

ldapBase

Base (node) from which to start LDAP searches.

If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

ldapBindDn

Full distinguished name (DN) that is used to bind to the LDAP server.

Default: cn=Manager,dc=netscaler,dc=com

ldapBindDnPassword

Password used to bind to the LDAP server.

ldapLoginName

LDAP login name attribute.

The NetScaler appliance uses the LDAP login name to query external LDAP servers or Active Directories.

searchFilter

String to be combined with the default LDAP user search string to form the search value. For example, if the search filter ?"vpnallowed=true"? is combined with the LDAP login name ?"samaccount"? and the user-supplied username is ?"bob"?, the result is the LDAP search string ""(&(vpnallowed=true)(samaccount=bob)"" (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.).

groupAttrName

LDAP group attribute name.

Used for group extraction on the LDAP server.

subAttributeName

LDAP group sub-attribute name.

Used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

Possible values: PLAINTEXT, TLS, SSL

Default value: AAA_LDAP_PLAINTEXT

svrType

The type of LDAP server.

Possible values: AD, NDS

Default value: AAA_LDAP_SERVER_TYPE_DEFAULT

ssoNameAttribute

LDAP single signon (SSO) attribute.

The NetScaler appliance uses the SSO name attribute to query external LDAP servers or Active Directories for an alternate username.

authentication

Perform LDAP authentication.

If authentication is disabled, any LDAP authentication attempt returns authentication success if the user is found.

CAUTION! Authentication should be disabled only for authorization group extraction or where other (non-LDAP) authentication methods are in use and either bound to a primary list or flagged as secondary.

Possible values: ENABLED, DISABLED

Default value: ENABLED

requireUser

Require a successful user search for authentication.

Possible values: YES, NO

Default value: YES

passwdChange

Allow password change requests.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nestedGroupExtraction

Allow nested group extraction, in which the NetScaler appliance queries external LDAP servers to determine whether a group is part of another group.

Possible values: ON, OFF

Default value: OFF

maxNestingLevel

If nested group extraction is ON, specifies the number of levels up to which group extraction is performed.

Default value: 2

Minimum value: 2

followReferrals

Setting this option to ON enables following LDAP referrals received from the LDAP server.

Possible values: ON, OFF

Default value: OFF

maxLDAPReferrals

Specifies the maximum number of nested referrals to follow.

Default value: 1

Minimum value: 1

validateServerCert

When to validate LDAP server certs

Possible values: YES, NO

Default value: NO

ldapHostname

Hostname for the LDAP server. If -validateServerCert is ON then this must be the host name on the certificate from the LDAP server.

A hostname mismatch will cause a connection failure.

groupNameIdentifier

Name that uniquely identifies a group in LDAP or Active Directory.

groupSearchAttribute

LDAP group search attribute.

Used to determine to which groups a group belongs.

groupSearchSubAttribute

LDAP group search subattribute.

Used to determine to which groups a group belongs.

groupSearchFilter

String to be combined with the default LDAP group search string to form the search value. For example, the group search filter ""vpnallowed=true"" when combined with the group identifier ""samaccount"" and the group name ""g1"" yields the LDAP search string ""(&(vpnallowed=true)(samaccount=g1)"". (Be sure to enclose the search string

in two sets of double quotation marks; both sets are needed.)

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

---

rm authentication ldapAction

Removes an LDAP profile (action). NOTE: An action cannot be removed if it is bound to a policy.

Synopsys

rm authentication ldapAction <name>

Arguments

name

Name of the LDAP profile (action) to be removed.

---

set authentication ldapAction

Modifies an LDAP server profile (action.) The profile contains all configuration data needed to communicate with that LDAP server.

Synopsys

set authentication ldapAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-ldapLoginName <string>] [-searchFilter <string>] [-groupAttrName <string>] [-subAttributeName <string>] [-secType <secType>] [-svrType (AD | NDS)] [-ssoNameAttribute <string>] [-authentication (ENABLED | DISABLED)] [-requireUser (YES | NO)] [-passwdChange (ENABLED | DISABLED)] [-validateServerCert (YES | NO)] [-ldapHostname <string>] [-nestedGroupExtraction (ON | OFF)] [-maxNestingLevel <positive_integer>] [-groupNameIdentifier <string>] [-groupSearchAttribute <string> [-groupSearchSubAttribute <string>]] [-groupSearchFilter <string>] [-followReferrals (ON | OFF)] [-maxLDAPReferrals <positive_integer>] [-defaultAuthenticationGroup <string>]

Arguments

name

Name of the LDAP profile to modify.

serverIP

IP address assigned to the LDAP server.

serverPort

Port on which the LDAP server accepts connections.

Default value: 389

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

ldapBase

Base (node) from which to start LDAP searches.

If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

ldapBindDn

Full distinguished name (DN) that is used to bind to the LDAP server.

Default: cn=Manager,dc=netscaler,dc=com

ldapBindDnPassword

Password used to bind to the LDAP server.

ldapLoginName

LDAP login name attribute.

The NetScaler appliance uses the LDAP login name to query external LDAP servers or Active Directories.

searchFilter

String to be combined with the default LDAP user search string to form the search value. For example, if the search filter ?"vpnallowed=true"? is combined with the LDAP login name ?"samaccount"? and the user-supplied username is ?"bob"?, the result is the LDAP search string ""(&(vpnallowed=true)(samaccount=bob)"" (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.).

groupAttrName

LDAP group attribute name.

Used for group extraction on the LDAP server.

subAttributeName

LDAP group sub-attribute name.

Used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

Possible values: PLAINTEXT, TLS, SSL

Default value: AAA_LDAP_PLAINTEXT

svrType

The type of LDAP server.

Possible values: AD, NDS

Default value: AAA_LDAP_SERVER_TYPE_DEFAULT

ssoNameAttribute

LDAP single signon (SSO) attribute.

The NetScaler appliance uses the SSO name attribute to query external LDAP servers or Active Directories for an alternate username.

authentication

Perform LDAP authentication.

If authentication is disabled, any LDAP authentication attempt returns authentication success if the user is found.

CAUTION! Authentication should be disabled only for authorization group extraction or where other (non-LDAP) authentication methods are in use and either bound to a primary list or flagged as secondary.

Possible values: ENABLED, DISABLED

Default value: ENABLED

requireUser

Require a successful user search for authentication.

Possible values: YES, NO

Default value: YES

passwdChange

Allow password change requests.

Possible values: ENABLED, DISABLED

Default value: DISABLED

validateServerCert

When to validate LDAP server certs

Possible values: YES, NO

Default value: NO

ldapHostname

Hostname for the LDAP server. If -validateServerCert is ON then this must be the host name on the certificate from the LDAP server.

A hostname mismatch will cause a connection failure.

nestedGroupExtraction

Allow nested group extraction, in which the NetScaler appliance queries external LDAP servers to determine whether a group is part of another group.

Possible values: ON, OFF

Default value: OFF

followReferrals

Setting this option to ON enables following LDAP referrals received from the LDAP server.

Possible values: ON, OFF

Default value: OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

---

unset authentication ldapAction

Use this command to remove authentication ldapAction settings.Refer to the set authentication ldapAction command for meanings of the arguments.

Synopsys

unset authentication ldapAction <name> [-serverIP] [-serverPort] [-authTimeout] [-ldapBase] [-ldapBindDn] [-ldapBindDnPassword] [-ldapLoginName] [-searchFilter] [-groupAttrName] [-subAttributeName] [-secType] [-svrType] [-ssoNameAttribute] [-authentication] [-requireUser] [-passwdChange] [-validateServerCert] [-ldapHostname] [-nestedGroupExtraction] [-maxNestingLevel] [-groupNameIdentifier] [-groupSearchAttribute] [-groupSearchSubAttribute] [-groupSearchFilter] [-followReferrals] [-maxLDAPReferrals] [-defaultAuthenticationGroup]

---

show authentication ldapAction

Displays the current configuration settings for the specified LDAP profile (action).

Synopsys

show authentication ldapAction [<name>]

Arguments

name

Name of the LDAP profile.

summary

fullValues

format

level

Outputs

serverIP

IP address assigned to the LDAP server.

serverPort

Port on which the LDAP server accepts connections.

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

ldapBindDn

Full distinguished name (DN) that is used to bind to the LDAP server.

Default: cn=Manager,dc=netscaler,dc=com

ldapBindDnPassword

Password used to bind to the LDAP server.

ldapLoginName

LDAP login name attribute.

The NetScaler appliance uses the LDAP login name to query external LDAP servers or Active Directories.

ldapBase

Base (node) from which to start LDAP searches.

If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

searchFilter

String to be combined with the default LDAP user search string to form the search value.

For example, if the search filter ?"vpnallowed=true"? is combined with the LDAP login name ?"samaccount"? and the user-supplied username is ?"bob"?, the result is the LDAP search string ""(&(vpnallowed=true)(samaccount=bob)"" (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.).

groupAttrName

LDAP group attribute name.

Used for group extraction on the LDAP server.

subAttributeName

LDAP group sub-attribute name.

Used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

svrType

The type of LDAP server.

ssoNameAttribute

LDAP single signon (SSO) attribute.

The NetScaler appliance uses the SSO name attribute to query external LDAP servers or Active Directories for an alternate username.

authentication

Perform LDAP authentication.

If authentication is disabled, any LDAP authentication attempt returns authentication success if the user is found.

CAUTION! Authentication should be disabled only for authorization group extraction or where other (non-LDAP) authentication methods are in use and either bound to a primary list or flagged as secondary.

requireUser

Require a successful user search for authentication.

Success

Failure

stateflag

nestedGroupExtraction

Allow nested group extraction, in which the NetScaler appliance queries external LDAP servers to determine whether a group is part of another group.

maxNestingLevel

If nested group extraction is ON, specifies the number of levels up to which group extraction is performed.

followReferrals

Setting this option to ON enables following LDAP referrals received from the LDAP server.

maxLDAPReferrals

Specifies the maximum number of nested referrals to follow.

validateServerCert

When to validate LDAP server certs

ldapHostname

Hostname for the LDAP server. If -validateServerCert is ON then this must be the host name on the certificate from the LDAP server.

A hostname mismatch will cause a connection failure.

groupNameIdentifier

Name that uniquely identifies a group in LDAP or Active Directory.

groupSearchAttribute

LDAP group search attribute.

Used to determine to which groups a group belongs.

groupSearchSubAttribute

LDAP group search subattribute.

Used to determine to which groups a group belongs.

groupSearchFilter

String to be combined with the default LDAP group search string to form the search value. For example, the group search filter ""vpnallowed=true"" when combined with the group identifier ""samaccount"" and the group name ""g1"" yields the LDAP search string ""(&(vpnallowed=true)(samaccount=g1)"". (Be sure

to enclose the search string in two sets of double quotation marks; both sets are needed.)

passwdChange

Allow password change requests.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

devno

count

authentication ldapPolicy

Sep 22, 2015

The following operations can be performed on "authentication ldapPolicy":

add | rm | set | unset | show

add authentication ldapPolicy

Adds an LDAP authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified LDAP server.

Synopsys

add authentication ldapPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the LDAP policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after LDAP policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the LDAP server.

reqAction

Name of the LDAP action to perform if the policy matches.

---

rm authentication ldapPolicy

Removes an LDAP policy.

Synopsys

rm authentication ldapPolicy <name>

Arguments

name

Name of the LDAP policy to remove.

---

set authentication ldapPolicy

Configures the specified LDAP policy.

Synopsys

set authentication ldapPolicy <name> [-rule <string>] [-reqAction <string>]

Arguments

name

Name of the LDAP policy.

rule

The new rule to associate with the policy.

reqAction

The new LDAP action to associate with the policy.

---

unset authentication ldapPolicy

Use this command to remove authentication ldapPolicy settings.Refer to the set authentication ldapPolicy command for meanings of the arguments.

Synopsys

unset authentication ldapPolicy <name> [-rule] [-reqAction]

---

show authentication ldapPolicy

Displays the current settings for the specified LDAP policy. If no policy name is provided, displays a list of all LDAP policies currently configured on the NetScaler appliance.

Synopsys

show authentication ldapPolicy [<name>]

Arguments

name

Name of the LDAP policy.

summary

fullValues

format

level

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the LDAP server.

reqAction

Name of the LDAP action to perform if the policy matches.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication localPolicy

Sep 22, 2015

The following operations can be performed on "authentication localPolicy":

add | rm | set | unset | show

add authentication localPolicy

Adds a policy for the NetScaler appliance to locally authenticate a user. The policy contains criteria that specify when and how to authenticate a user.

Synopsys

add authentication localPolicy <name> <rule>

Arguments

name

Name for the local authentication policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after local policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

---

rm authentication localPolicy

Removes the specified local authentication policy.

Synopsys

rm authentication localPolicy <name>

Arguments

name

Name of the local policy to remove.

---

set authentication localPolicy

Configures the specified local authentication policy.

Synopsys

set authentication localPolicy <name> -rule <expression>

Arguments

name

Name of the local authentication policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

---

unset authentication localPolicy

Use this command to remove authentication localPolicy settings.Refer to the set authentication localPolicy command for meanings of the arguments.

Synopsys

unset authentication localPolicy <name> -rule

---

show authentication localPolicy

Displays the current settings for the specified local authentication policy. If no policy name is provided, displays a list of all local authentication policies currently configured on the NetScaler appliance.

Synopsys

show authentication localPolicy [<name>]

Arguments

name

Name of the local authentication policy.

summary

fullValues

format

level

Outputs

rule

The new rule associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

reqAction

The name of the RADIUS action the policy uses

bindPolicyType

policyType

devno

count

stateflag

authentication negotiateAction

Sep 22, 2015

The following operations can be performed on "authentication negotiateAction":

add | rm | set | unset | show

add authentication negotiateAction

Creates an action (profile) for an Active Directory (AD) server that is used as a Kerberos Key Distribution Center (KDC). The profile contains all configuration data necessary to communicate with that AD KDC server.

Synopsys

add authentication negotiateAction <name> {-domain <string>} {-domainUser <string>} {-domainUserPasswd } {-OU <string>} [-defaultAuthenticationGroup <string>] [-keytab <string>]

Arguments

name

Name for the AD KDC server profile (negotiate action).

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after AD KDC server profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

domain

Domain name of the AD KDC server.

domainUser

User name that the NetScaler appliance uses to join the AD KDC server domain.

The NetScaler appliance uses the domain user name to check the health of the AD KDC server.

domainUserPasswd

Password that the NetScaler appliance uses to join the AD KDC server domain.

OU

Active Directory organizational units (OU) attribute.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

keytab

The path to the keytab file

---

rm authentication negotiateAction

Removes an AD KDC server profile (negotiate action). An action cannot be removed if it is bound to a policy.

Synopsys

rm authentication negotiateAction <name>

Arguments

name

Name of the AD KDC server profile to be removed.

---

set authentication negotiateAction

Configures an AD KDC server profile (negotiate action).

Synopsys

set authentication negotiateAction <name> [-domain <string>] [-domainUser <string>] [-domainUserPasswd] [-OU <string>] [-defaultAuthenticationGroup <string>] [-keytab <string>]

Arguments

name

Name of the AD KDC server profile.

domain

Domain name of the AD KDC server.

domainUser

User name that the NetScaler appliance uses to join the AD KDC server domain.

The NetScaler appliance uses the domain user name to check the health of the AD KDC server.

domainUserPasswd

Password that the NetScaler appliance uses to join the AD KDC server domain.

OU

Active Directory organizational units (OU) attribute.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

keytab

The path to the keytab file

---

unset authentication negotiateAction

Use this command to remove authentication negotiateAction settings.Refer to the set authentication negotiateAction command for meanings of the arguments.

Synopsys

unset authentication negotiateAction <name> [-domain] [-domainUser] [-domainUserPasswd] [-OU] [-defaultAuthenticationGroup]

---

show authentication negotiateAction

Displays the current configuration settings for the specified AD KDC server profile (negotiate action).

Synopsys

show authentication negotiateAction [<name>]

Arguments

name

Name of the AD KDC server profile.

summary

fullValues

format

level

Outputs

domain

Domain name of the AD KDC server.

domainUser

User name that the NetScaler appliance uses to join the AD KDC server domain.

The NetScaler appliance uses the domain user name to check the health of the AD KDC server.

domainUserPasswd

Password that the NetScaler appliance uses to join the AD KDC server domain.

OU

Active Directory organizational units (OU) attribute.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

keytab

The path to the keytab file

kcdSPN

Host SPN extracted from keytab file.

stateflag

devno

count

authentication negotiatePolicy

Sep 22, 2015

The following operations can be performed on "authentication negotiatePolicy":

add | rm | set | unset | show

add authentication negotiatePolicy

Adds an Active Directory (AD) Kerberos Key Distribution Center (KCD) authentication policy (negotiate policy). The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified AD KCD server.

Synopsys

add authentication negotiatePolicy <name> <rule> <reqAction>

Arguments

name

Name for the negotiate authentication policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after AD KCD (negotiate) policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the AD KCD server.

reqAction

Name of the negotiate action to perform if the policy matches.

---

rm authentication negotiatePolicy

Removes the specified AD KCD (negotiate) policy.

Synopsys

rm authentication negotiatePolicy <name>

Arguments

name

Name of the negotiate policy to remove.

---

set authentication negotiatePolicy

Modifies the specified AD KCD (negotiate) policy.

Synopsys

set authentication negotiatePolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the negotiate policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the AD KCD server.

reqAction

Name of the negotiate action to perform if the policy matches.

---

unset authentication negotiatePolicy

Use this command to remove authentication negotiatePolicy settings.Refer to the set authentication negotiatePolicy command for meanings of the arguments.

Synopsys

unset authentication negotiatePolicy <name> [-rule] [-reqAction]

---

show authentication negotiatePolicy

Displays the current settings for the specified AD KCD (negotiate) policy. If no policy name is provided, displays a list of all negotiate policies currently configured on the NetScaler appliance.

Synopsys

show authentication negotiatePolicy [<name>]

Arguments

name

Name of the negotiate policy.

summary

fullValues

format

level

Outputs

rule

The name of the new rule associated with the policy.

reqAction

The name of the Negotiate action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication radiusAction

Sep 22, 2015

The following operations can be performed on "authentication radiusAction":

add | rm | set | unset | show

add authentication radiusAction

Creates an action (profile) for a RADIUS server. The profile contains all configuration data necessary to communicate with that RADIUS server.

Synopsys

add authentication radiusAction <name> {-serverIP <ip_addr|ipv6_addr|*>} [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip (ENABLED | DISABLED)] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting (ON | OFF)] [-pwdVendorID <positive_integer> [-pwdAttributeType <positive_integer>]] [-defaultAuthenticationGroup <string>] [-callingstationid (ENABLED | DISABLED)]

Arguments

name

Name for the RADIUS action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the RADIUS action is added.

serverIP

IP address assigned to the RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

Default value: 1812

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

radKey

Key shared between the RADIUS server and the NetScaler appliance.

Required to allow the NetScaler appliance to communicate with the RADIUS server.

radNASip

If enabled, the NetScaler appliance IP address (NSIP) is sent to the RADIUS server as the Network Access Server IP (NASIP) address.

The RADIUS protocol defines the meaning and use of the NASIP address.

Possible values: ENABLED, DISABLED

radNASid

If configured, this string is sent to the RADIUS server as the Network Access Server ID (NASID).

radVendorID

RADIUS vendor ID attribute, used for RADIUS group extraction.

Minimum value: 1

radAttributeType

RADIUS attribute type, used for RADIUS group extraction.

Minimum value: 1

radGroupsPrefix

RADIUS groups prefix string.

This groups prefix precedes the group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

RADIUS group separator string

The group separator delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Encoding type for passwords in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

Possible values: pap, chap, mschapv1, mschapv2

Default value: AAA_PAP

ipVendorID

Vendor ID of the intranet IP attribute in the RADIUS response.

NOTE: A value of 0 indicates that the attribute is not vendor encoded.

ipAttributeType

Remote IP address attribute type in a RADIUS response.

Minimum value: 1

accounting

Whether the RADIUS server is currently accepting accounting messages.

Possible values: ON, OFF

pwdVendorID

Vendor ID of the attribute, in the RADIUS response, used to extract the user password.

Minimum value: 1

pwdAttributeType

Vendor-specific password attribute type in a RADIUS response.

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Possible values: ENABLED, DISABLED

Default value: DISABLED

---

rm authentication radiusAction

Removes a RADIUS profile (action). An action cannot be removed as long as it is bound to a policy.

Synopsys

rm authentication radiusAction <name>

Arguments

name

Name of the action to be removed.

---

set authentication radiusAction

Configures a RADIUS server profile (action). The profile contains all configuration data needed to communicate with that RADIUS server.

Synopsys

set authentication radiusAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip (ENABLED | DISABLED)] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting (ON | OFF)] [-pwdVendorID <positive_integer>] [-pwdAttributeType <positive_integer>] [-defaultAuthenticationGroup <string>] [-callingstationid (ENABLED | DISABLED)]

Arguments

name

Name of the RADIUS profile.

serverIP

IP address assigned to the RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

Default value: 1812

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

radKey

Key shared between the RADIUS server and the NetScaler appliance.

Required to allow the NetScaler appliance to communicate with the RADIUS server.

radNASip

If enabled, the NetScaler appliance IP address (NSIP) is sent to the RADIUS server as the Network Access Server IP (NASIP) address.

The RADIUS protocol defines the meaning and use of the NASIP address.

Possible values: ENABLED, DISABLED

radNASid

If configured, this string is sent to the RADIUS server as the Network Access Server ID (NASID).

radVendorID

RADIUS vendor ID attribute, used for RADIUS group extraction.

Minimum value: 1

radAttributeType

RADIUS attribute type, used for RADIUS group extraction.

Minimum value: 1

radGroupsPrefix

RADIUS groups prefix string.

This groups prefix precedes the group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

RADIUS group separator string

The group separator delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Encoding type for passwords in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

Possible values: pap, chap, mschapv1, mschapv2

Default value: AAA_PAP

ipVendorID

Vendor ID of the intranet IP attribute in the RADIUS response.

NOTE: A value of 0 indicates that the attribute is not vendor encoded.

ipAttributeType

Remote IP address attribute type in a RADIUS response.

Minimum value: 1

accounting

Whether the RADIUS server is currently accepting accounting messages.

Possible values: ON, OFF

pwdVendorID

Vendor ID of the attribute, in the RADIUS response, used to extract the user password.

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Possible values: ENABLED, DISABLED

Default value: DISABLED

---

unset authentication radiusAction

Use this command to remove authentication radiusAction settings.Refer to the set authentication radiusAction command for meanings of the arguments.

Synopsys

unset authentication radiusAction <name> [-serverIP] [-serverPort] [-authTimeout] [-radNASip] [-radNASid] [-radVendorID] [-radAttributeType] [-radGroupsPrefix] [-radGroupSeparator] [-passEncoding] [-ipVendorID] [-ipAttributeType] [-accounting] [-pwdVendorID] [-pwdAttributeType] [-defaultAuthenticationGroup] [-callingstationid]

---

show authentication radiusAction

Displays the current configuration settings for the specified RADIUS profile (action).

Synopsys

show authentication radiusAction [<name>]

Arguments

name

Name of the RADIUS profile.

summary

fullValues

format

level

Outputs

serverIP

IP address assigned to the RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

radKey

Key shared between the RADIUS server and the NetScaler appliance.

Required to allow the NetScaler appliance to communicate with the RADIUS server.

radNASip

If enabled, the NetScaler appliance IP address (NSIP) is sent to the RADIUS server as the Network Access Server IP (NASIP) address.

The RADIUS protocol defines the meaning and use of the NASIP address.

IPAddress

IP address.

radNASid

If configured, this string is sent to the RADIUS server as the Network Access Server ID (NASID).

radVendorID

RADIUS vendor ID attribute, used for RADIUS group extraction.

radAttributeType

RADIUS attribute type, used for RADIUS group extraction.

radGroupsPrefix

RADIUS groups prefix string.

This groups prefix precedes the group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

RADIUS group separator string

The group separator delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Encoding type for passwords in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

ipVendorID

Vendor ID of the intranet IP attribute in the RADIUS response.

NOTE: A value of 0 indicates that the attribute is not vendor encoded.

ipAttributeType

Remote IP address attribute type in a RADIUS response.

accounting

Whether the RADIUS server is currently accepting accounting messages.

Success

Failure

stateflag

pwdVendorID

Vendor ID of the attribute, in the RADIUS response, used to extract the user password.

pwdAttributeType

Vendor-specific password attribute type in a RADIUS response.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

devno

count

authentication radiusPolicy

Sep 22, 2015

The following operations can be performed on "authentication radiusPolicy":

add | rm | set | unset | show

add authentication radiusPolicy

Adds a RADIUS authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the RADIUS server.

Synopsys

add authentication radiusPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the RADIUS authentication policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after RADIUS policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the RADIUS server.

reqAction

Name of the RADIUS action to perform if the policy matches.

---

rm authentication radiusPolicy

Removes a RADIUS authentication policy.

Synopsys

rm authentication radiusPolicy <name>

Arguments

name

Name of the RADIUS authentication policy to remove.

---

set authentication radiusPolicy

Configures the specified RADIUS authentication policy.

Synopsys

set authentication radiusPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the RADIUS authentication policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the RADIUS server.

reqAction

Name of the RADIUS action to perform if the policy matches.

---

unset authentication radiusPolicy

Use this command to remove authentication radiusPolicy settings.Refer to the set authentication radiusPolicy command for meanings of the arguments.

Synopsys

unset authentication radiusPolicy <name> [-rule] [-reqAction]

---

show authentication radiusPolicy

Displays the current settings for the specified RADIUS authentication policy. If no policy name is provided, displays a list of all RADIUS authentication policies currently configured on the NetScaler appliance.

Synopsys

show authentication radiusPolicy [<name>]

Arguments

name

Name of the RADIUS authentication policy.

summary

fullValues

format

level

Outputs

rule

The new rule associated with the policy.

reqAction

The new RADIUS action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication samlAction

Sep 22, 2015

The following operations can be performed on "authentication samlAction":

add | rm | set | unset | show

add authentication samlAction

Creates an action (profile) for a Security Assertion Markup Language (SAML) server. The profile contains all configuration data necessary to communicate with that SAML server.

Synopsys

add authentication samlAction <name> {-samlIdPCertName <string>} {-samlSigningCertName <string>} {-samlRedirectUrl <string>} {-samlUserField <string>} {-samlRejectUnsignedAssertion (ON | OFF)} {-samlIssuerName <string>} {-samlTwoFactor (ON | OFF)} [-defaultAuthenticationGroup <string>]

Arguments

name

Name for the SAML server profile (action).

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after SAML profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

samlIdPCertName

Name of the SAML server as given in that server?s SSL certificate.

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

samlRedirectUrl

URL to which users are redirected after successful authentication.

samlUserField

SAML user ID, as given in the SAML assertion.

samlRejectUnsignedAssertion

Reject unsigned SAML assertions.

Possible values: ON, OFF

Default value: ON

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

samlTwoFactor

Option to enable second factor after SAML

Possible values: ON, OFF

Default value: NS_OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

---

rm authentication samlAction

Removes a SAML profile (action). An action cannot be removed if it is bound to a policy.

Synopsys

rm authentication samlAction <name>

Arguments

name

Name of the SAML profile to be removed.

---

set authentication samlAction

Modifies the specified parameters of a SAML server profile (action).

Synopsys

set authentication samlAction <name> [-samlIdPCertName <string>] [-samlSigningCertName <string>] [-samlRedirectUrl <string>] [-samlUserField <string>] [-samlRejectUnsignedAssertion (ON | OFF)] [-samlIssuerName <string>] [-samlTwoFactor (ON | OFF)] [-defaultAuthenticationGroup <string>]

Arguments

name

Name of the SAML profile (action) to modify.

samlIdPCertName

Name of the SAML server as given in that server?s SSL certificate.

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

samlRedirectUrl

URL to which users are redirected after successful authentication.

samlUserField

SAML user ID, as given in the SAML assertion.

samlRejectUnsignedAssertion

Reject unsigned SAML assertions.

Possible values: ON, OFF

Default value: ON

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

samlTwoFactor

Option to enable second factor after SAML

Possible values: ON, OFF

Default value: NS_OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

---

unset authentication samlAction

Use this command to remove authentication samlAction settings.Refer to the set authentication samlAction command for meanings of the arguments.

Synopsys

unset authentication samlAction <name> [-samlIdPCertName] [-samlSigningCertName] [-samlRedirectUrl] [-samlUserField] [-samlRejectUnsignedAssertion] [-samlIssuerName] [-samlTwoFactor] [-defaultAuthenticationGroup]

---

show authentication samlAction

Displays the current configuration settings for the specified SAML server profile (action).

Synopsys

show authentication samlAction [<name>]

Arguments

name

Name of the SAML server profile.

summary

fullValues

format

level

Outputs

samlIdPCertName

Name of the SAML server as given in that server?s SSL certificate.

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

samlRedirectUrl

URL to which users are redirected after successful authentication.

samlUserField

SAML user ID, as given in the SAML assertion.

samlRejectUnsignedAssertion

Reject unsigned SAML assertions.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

samlTwoFactor

Option to enable second factor after SAML

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to

extracted groups.

devno

count

stateflag

authentication samlPolicy

Sep 22, 2015

The following operations can be performed on "authentication samlPolicy":

add | rm | set | unset | show

add authentication samlPolicy

Adds a SAML authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified SAML server.

Synopsys

add authentication samlPolicy <name> <rule> <reqAction>

Arguments

name

Name for the SAML policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after SAML policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the SAML server.

reqAction

Name of the SAML authentication action to be performed if the policy matches.

---

rm authentication samlPolicy

Removes the specified SAML policy.

Synopsys

rm authentication samlPolicy <name>

Arguments

name

Name of the policy to remove.

---

set authentication samlPolicy

Modifies the specified parameters of a SAML policy.

Synopsys

set authentication samlPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the SAML policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the SAML server.

reqAction

Name of the SAML authentication action to be performed if the policy matches.

---

unset authentication samlPolicy

Use this command to remove authentication samlPolicy settings.Refer to the set authentication samlPolicy command for meanings of the arguments.

Synopsys

unset authentication samlPolicy <name> [-rule] [-reqAction]

---

show authentication samlPolicy

Displays the current settings for the specified SAML policy. If no policy name is provided, displays a list of all SAML policies currently configured on the NetScaler appliance.

Synopsys

show authentication samlPolicy [<name>]

Arguments

name

Name of the SAML policy.

summary

fullValues

format

level

Outputs

rule

The name of the new rule associated with the policy.

reqAction

The name of the SAML action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication tacacsAction

Sep 22, 2015

The following operations can be performed on "authentication tacacsAction":

add | rm | set | unset | show

add authentication tacacsAction

Creates an action (profile) for a TACACS+ server. The profile contains all configuration data necessary to communicate with that TACACS+ server.

Synopsys

add authentication tacacsAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-tacacsSecret } [-authorization (ON | OFF)] [-accounting (ON | OFF)] [-auditFailedCmds (ON | OFF)] [-defaultAuthenticationGroup <string>]

Arguments

name

Name for the TACACS+ profile (action).

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after TACACS profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

serverIP

IP address assigned to the TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

Default value: 49

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the TACACS+ server.

Default value: 3

Minimum value: 1

tacacsSecret

Key shared between the TACACS+ server and the NetScaler appliance.

Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

Possible values: ON, OFF

accounting

Whether the TACACS+ server is currently accepting accounting messages.

Possible values: ON, OFF

auditFailedCmds

The state of the TACACS+ server that will receive accounting messages.

Possible values: ON, OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

---

rm authentication tacacsAction

Removes a TACACS+ profile (action). A profile cannot be removed as long as it is bound to a policy.

Synopsys

rm authentication tacacsAction <name>

Arguments

name

Name of the profile to be removed.

---

set authentication tacacsAction

Modifies a TACACS+ server profile (action).

Synopsys

set authentication tacacsAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-tacacsSecret } [-authorization (ON | OFF)] [-accounting (ON | OFF)] [-auditFailedCmds (ON |

OFF)] [-defaultAuthenticationGroup <string>]

Arguments

name

Name of the TACACS+ profile to modify.

serverIP

IP address assigned to the TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

Default value: 49

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the TACACS+ server.

Default value: 3

Minimum value: 1

tacacsSecret

Key shared between the TACACS+ server and the NetScaler appliance.

Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

Possible values: ON, OFF

accounting

Whether the TACACS+ server is currently accepting accounting messages.

Possible values: ON, OFF

auditFailedCmds

The state of the TACACS+ server that will receive accounting messages.

Possible values: ON, OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

---

unset authentication tacacsAction

Use this command to remove authentication tacacsAction settings.Refer to the set authentication tacacsAction command for meanings of the arguments.

Synopsys

unset authentication tacacsAction <name> [-serverIP] [-serverPort] [-authTimeout] [-tacacsSecret] [-authorization] [-accounting] [-auditFailedCmds] [-defaultAuthenticationGroup]

---

show authentication tacacsAction

Displays the current configuration settings for the specified TACACS+ profile (action).

Synopsys

show authentication tacacsAction [<name>]

Arguments

name

Name of the TACACS+ profile.

summary

fullValues

format

level

Outputs

serverIP

IP address assigned to the TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

authTimeout

Number of seconds the NetScaler appliance waits for a response from the TACACS+ server.

tacacsSecret

Key shared between the TACACS+ server and the NetScaler appliance.

Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

accounting

Whether the TACACS+ server is currently accepting accounting messages.

auditFailedCmds

The state of the TACACS+ server that will receive accounting messages.

Success

Failure

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

stateflag

devno

count

authentication tacacsPolicy

Sep 22, 2015

The following operations can be performed on "authentication tacacsPolicy":

add | rm | set | unset | show

add authentication tacacsPolicy

Adds a TACACS+ authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified TACACS+ server.

Synopsys

add authentication tacacsPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the TACACS+ policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after TACACS+ policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the TACACS+ server.

reqAction

Name of the TACACS+ action to perform if the policy matches.

---

rm authentication tacacsPolicy

Removes the specified TACACS+ policy.

Synopsys

rm authentication tacacsPolicy <name>

Arguments

name

Name of the TACACS+ policy to remove.

---

set authentication tacacsPolicy

Configures the specified TACACS+ policy.

Synopsys

set authentication tacacsPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the TACACS+ policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the TACACS+ server.

reqAction

Name of the TACACS+ action to perform if the policy matches.

---

unset authentication tacacsPolicy

Use this command to remove authentication tacacsPolicy settings.Refer to the set authentication tacacsPolicy command for meanings of the arguments.

Synopsys

unset authentication tacacsPolicy <name> [-rule] [-reqAction]

---

show authentication tacacsPolicy

Displays the current settings for the specified TACACS+ policy. If no policy name is provided, displays a list of all TACACS+ policies currently configured on the NetScaler appliance.

Synopsys

show authentication tacacsPolicy [<name>]

Arguments

name

Name of the TACACS+ policy.

summary

fullValues

format

level

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the TACACS+ server.

reqAction

Name of the TACACS+ action to perform if the policy matches.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication vserver

Sep 22, 2015

The following operations can be performed on "authentication vserver":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

add authentication vserver

Creates an authentication virtual server.

Synopsys

add authentication vserver <name> <serviceType> (<IPAddress> [-range <positive_integer>]) <port> [-state (ENABLED | DISABLED)] [-authentication (ON | OFF)] [-AuthenticationDomain <string>] [-comment <string>] [-td <positive_integer>] [-appflowLog (ENABLED | DISABLED)] [-maxLoginAttempts <positive_integer> [-failedLoginTimeout <positive_integer>]]

Arguments

name

Name for the new authentication virtual server.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the authentication virtual server is added by using the rename authentication vserver command.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

serviceType

Protocol type of the authentication virtual server. Always SSL.

Possible values: SSL

Default value: NSSVC_SSL

IPAddress

IP address of the authentication virtual server, if a single IP address is assigned to the virtual server.

port

TCP port on which the virtual server accepts connections.

Minimum value: 1

state

Initial state of the new virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

authentication

Require users to be authenticated before sending traffic through this virtual server.

Possible values: ON, OFF

Default value: ON

AuthenticationDomain

Fully-qualified domain name (FQDN) of the authentication virtual server.

comment

Any comments associated with this virtual server.

td

Traffic Domain ID

Maximum value: 4094

appflowLog

Log AppFlow flow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

Maximum value: 255

Example

The following example creates an authentication vserver named myauthenticationvip which supports SSL portocol and with AAA functionality enabled: vserver myauthent

---

rm authentication vserver

Removes an authentication virtual server.

Synopsys

rm authentication vserver <name>@ ...

Arguments

name

Name of the authentication virtual server to remove.

Example

rm vserver authn_vip

---

set authentication vserver

Modifies the specified parameters of an existing authentication virtual server.

Synopsys

set authentication vserver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-authentication (ON | OFF)] [-AuthenticationDomain <string>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-maxLoginAttempts <positive_integer>] [-failedLoginTimeout <positive_integer>]

Arguments

name

Name of the virtual server to modify.

IPAddress

IP address of the authentication virtual server, if a single IP address is assigned to the virtual server.

authentication

Require users to be authenticated before sending traffic through this virtual server.

Possible values: ON, OFF

Default value: ON

AuthenticationDomain

Fully-qualified domain name (FQDN) of the authentication virtual server.

comment

Any comments associated with this virtual server.

appflowLog

Log AppFlow flow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

Maximum value: 255

failedLoginTimeout

Failed Login timeout

Minimum value: 1

---

unset authentication vserver

Removes the settings of an existing authentication virtual server. Attributes for which a default value is available revert to their default values. Refer to the set authentication vserver command for descriptions of the parameters..Refer to the set authentication vserver command for meanings of the arguments.

Synopsys

unset authentication vserver <name> [-AuthenticationDomain] [-maxLoginAttempts] [-authentication] [-comment] [-appflowLog]

---

bind authentication vserver

Binds authentication policies to an authentication virtual server.

Synopsys

bind authentication vserver <name> [-policy <string> [-priority <positive_integer>] [-secondary] [-groupExtraction]]

Arguments

name

Name of the authentication virtual server to which to bind the policy.

policy

Name of the policy to bind to the virtual server.

---

unbind authentication vserver

Unbinds the specified policy from the specified authentication virtual server.

Synopsys

unbind authentication vserver <name> [-policy <string> [-secondary] [-groupExtraction]]

Arguments

name

Name of the virtual server.

policy

Name of the policy to be unbound.

---

enable authentication vserver

Enables an authentication virtual server that is disabled. Note: Virtual servers, when added, are normally enabled by default.

Synopsys

enable authentication vserver <name>@

Arguments

name

Name of the virtual server to enable.

Example

enable vserver authentication1

---

disable authentication vserver

Disables an authentication virtual server, taking it out of service.

Synopsys

disable authentication vserver <name>@

Arguments

name

Name of the virtual server to disable.

Notes:

1. The NetScaler appliance still responds to ARP and/or ping requests for the IP address of disabled virtual servers.

2. Because the virtual server configuration still exists on the NetScaler appliance, you can reenable the virtual server.

Example

disable vserver authn_vip

---

show authentication vserver

Displays the configuration of the specified authentication virtual server. If no authentication virtual server is specified, displays a list of all authentication virtual servers that are currently configured on the NetScaler appliance.

Synopsys

show authentication vserver [<name>] show authentication vserver stats - alias for 'stat authentication vserver'

Arguments

name

Name of the authentication virtual server.

summary

fullValues

format

level

Outputs

IPAddress

The IP address of the authentication server.

td

Traffic Domain ID

value

Indicates whether or not the certificate is bound or if SSL offload is disabled.

port

The virtual TCP port of the authentication vserver.

range

The range of authentication vserver IP addresses. The new range of authentication vservers will have IP addresses consecutively numbered, starting with the primary address specified with the <ipaddress> argument.

serviceType

The authentication vserver's protocol type, Currently the only possible value is SSL.

type

The type of Virtual Server, e.g. CONTENT based or ADDRESS based.

state

Initial state of the new virtual server.

status

Whether or not this vserver responds to ARPs and whether or not round-robin selection is temporarily in effect.

cacheType

Virtual server's cache type. The options are: TRANSPARENT, REVERSE and FORWARD.

redirect

The cache redirect policy.

The valid redirect policies are:

l. CACHE - Directs all requests to the cache.

2. POLICY - Applies cache redirection policy to determine whether the request should be directed to the cache or origin. This is the default setting.

3. ORIGIN - Directs all requests to the origin server.

precedence

This argument is used only when configuring content switching on the specified virtual server. This is applicable only

if both the URL and RULE-based policies have been configured on the same virtual server.

It specifies the type of policy (URL or RULE) that takes precedence on the content switching virtual server. The default setting is RULE.

l URL - In this case, the incoming request is matched against the URL-based policies before the rule-based policies.

l RULE - In this case, the incoming request is matched against the rule-based policies before the URL-based policies.

For all URL-based policies, the precedence hierarchy is:

1. Domain and exact URL

2. Domain, prefix and suffix

3. Domain and suffix

4. Domain and prefix

5. Domain only

6. Exact URL

7. Prefix and suffix

8. Suffix only

9. Prefix only

10. Default

redirectURL

The URL where traffic is redirected if the virtual server in system becomes unavailable. WARNING! Make sure that the domain you specify in the URL does not match the domain specified in the -d domainName argument of the ###add cs policy### command. If the same domain is specified in both arguments, the request will be continuously redirected to the same unavailable virtual server in the system. If so, the user may not get the requested content.

authentication

Indicates whether or not authentication is being applied to incoming users to the VPN.

curAAAUsers

The number of current users logged in to this vserver.

AuthenticationDomain

Fully-qualified domain name (FQDN) of the authentication virtual server.

rule

The name of the rule, or expression, if any, that policy for the authentication server is to use. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide. The default rule is ns_true.

policyName

The name of the policy, if any, bound to the authentication vserver.NOTE: This attribute is deprecated.Replaced by Policy field

policy

The name of the policy, if any, bound to the authentication vserver.

serviceName

The name of the service, if any, to which the vserver policy is bound.

weight

Weight for this service, if any. This weight is used when the system performs load balancing, giving greater priority to a specific service. It is useful when the services bound to a virtual server are of different capacity.

cacheVserver

The name of the default target cache virtual server, if any, to which requests are redirected.

backupVServer

The name of the backup vpn virtual server for this vpn virtual server.

cltTimeout

The idle time, if any, in seconds after which the client connection is terminated.

soMethod

VPN client applications are allocated from a block of Intranet IP addresses.

That block may be exhausted after a certain number of connections. This switch specifies the

method used to determine whether or not a new connection will spillover, or exhaust, the allocated block of

Intranet IP addresses for that application. Possible values are CONNECTION or DYNAMICCONNECTION.

CONNECTION means that a static integer value is the hard limit for the spillover threshold. The spillover

threshold is described below. DYNAMICCONNECTION means that the spillover threshold is set according to

the maximum number of connections defined for the vpn vserver.

soThreshold

VPN client applications are allocated from a block of Intranet IP addresses.

That block may be exhausted after a certain number of connections.

The value of this option is number of client connections after which the Mapped IP address is used

as the client source IP address instead of an address from the allocated block of Intranet IP addresses.

soPersistence

Whether or not cookie-based site persistance is enabled for this VPN vserver. Possible values are 'ConnectionProxy', HTTPRedirect, or NONE

soPersistenceTimeOut

The timeout, if any, for cookie-based site persistance of this VPN vserver.

priority

The priority, if any, of the vpn vserver policy.

downStateFlush

Perform delayed clean up of connections on this vserver.

actType

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

Listenpolicy

Listenpolicy configured for authentication vserver

Listenpriority

Priority of listen policy for authentication vserver

tcpProfileName

The name of the TCP profile.

httpProfileName

Name of the HTTP profile.

comment

Any comments associated with this virtual server.

policySubType

stateflag

flags

appflowLog

Log AppFlow flow information.

vstype

Virtual Server Type, e.g. Load Balancing, Content Switch, Cache Redirection

ngname

Nodegroup devno to which this lbvsever belongs to

maxLoginAttempts

Maximum Number of login Attempts

failedLoginTimeout

Failed Login timeout

secondary

Bind the authentication policy to the secondary chain.

Provides for multifactor authentication in which a user must authenticate via both a primary authentication method and, afterward, via a secondary authentication method.

Because user groups are aggregated across authentication systems, usernames must be the same on all authentication servers. Passwords can be different.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

devno

count

Example

show authentication vserver

---

stat authentication vserver

Displays statistics about the specified authentication virtual server. If no authentication virtual server is specified, displays statistics for all authentication virtual servers that are currently configured on the NetScaler appliance.

Synopsys

stat authentication vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the authentication virtual server.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

---

rename authentication vserver

Rename an authentication virtual server.

Synopsys

rename authentication vserver <name>@ <newName>@

Arguments

name

Current name of the authentication virtual server.

newName

New name of the authentication virtual server.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

Example

```
rename authentication vserver av1 av_new
```

Authorization Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- authorization action
- authorization policy
- authorization policylabel

authorization action

Sep 22, 2015

The following operations can be performed on "authorization action":

Show details of authorization actions.

show authorization action [<name>]

name

Name of authorization action

summary

fullValues

format

level

devno

count

stateflag

authorization policy

Sep 22, 2015

The following operations can be performed on "authorization policy":

add | rm | set | rename | show

Creates an authorization policy. Authorization policies allow AAA users and AAA groups to access resources through SSL VPN/AAA-TM enabled virtual servers.

add authorization policy <name> <rule> <action>

name

Name for the new authorization policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the authorization policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authorization policy? or ?my authorization policy?).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

action

Action to perform if the policy matches: either allow or deny the request.

Example: Consider the following authorization policy, "author-policy", add authorization policy author-policy "URL == /*.gif" DENY bind aaa user foo -policy author-pol

---

Removes an authorization policy.

rm authorization policy <name>

name

Name of the authorization policy to be removed.

Configures the specified parameters of an authorization policy.

set authorization policy <name> [-rule <expression>] [-action <string>]

name

Name of the authorization policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

action

Action to perform if the policy matches: either allow or deny the request.

Rename a author policy.

rename authorization policy <name>@ <newName>@

name

The name of the author policy.

newName

The new name of the author policy.

rename auth policy oldname newname

Displays the current settings for the specified authorization policy. If no policy name is provided, displays a list of all authorization policies currently configured on the NetScaler appliance.

show authorization policy [<name>]

name

Name of the authorization policy.

summary

fullValues

format

level

rule

Rule of the policy.

action

Authorization action associated with the policy. It can be either ALLOW or DENY.

boundTo

The entity name to which policy is bound

activePolicy

priority

flag

bindPolicyType

policyType

vserverType

devno

count

stateflag

---

authorization policylabel

Sep 22, 2015

The following operations can be performed on "authorization policylabel":

add | rm | bind | unbind | rename | show | stat

Creates a user-defined authorization policy label.

add authorization policylabel <labelName>

labelName

Name for the new authorization policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the authorization policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authorization policy label? or ?authorization policy label?).

add authorization policylabel trans_http_url

Removes an authorization policy label.

rm authorization policylabel <labelName>

labelName

Name of the authorization policy label to remove.

rm authorization policylabel trans_http_url

Binds an authorization policy to a label.

bind authorization policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

labelName

Name of the authorization policy label to which to bind the policy.

policyName

Name of the authorization policy to bind to the policy label.

i) bind authorization policylabel trans_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind authorization policylabel trans_http_url pol_2 2

Unbinds the specified policy from the specified authorization policy label.

unbind authorization policylabel <labelName> <policyName> [-priority <positive_integer>]

labelName

Name for the new authorization policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the authorization policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authorization policy label? or ?authorization policy label?).

policyName

Name of the authorization policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind authorization policylabel trans_http_url pol_1

Rename a auth policy label.

rename authorization policylabel <labelName>@ <newName>@

labelName

The name of the auth policy label

newName

The new name of the auth policy label

rename auth policy label oldname newname

Displays the current settings for the specified authorization policy label. If no policy name is provided, displays a list of all authorization policy labels currently configured on the NetScaler appliance.

show authorization policylabel [<labelName>]

labelName

Name of the authorization policy label.

summary

fullValues

format

level

stateflag

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the authorization policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of invocation. Available settings function as follows:

* reqvserver - Send the request to the specified request virtual server.

* resvserver - Send the response to the specified response virtual server.

* policylabel - Invoke the specified policy label.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

flowType

Flowtype of the bound authorization policy.

description

Description of the policylabel

flags

devno

count

i) show authorization policylabel trans_http_url ii) show authorization policylabel

---

Displays statistics for the specified authorization policy label. If no authorization policy label is specified, displays a list

of all authorization policy labels.

stat authorization policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

labelName

Name of the authorization policy label.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy Label Hits (Hits)

Number of times policy label was invoked.

AutoScale Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- autoscale action
- autoscale policy
- autoscale profile

autoscale action

Sep 22, 2015

The following operations can be performed on "autoscale action":

add | rm | set | unset | show

Create a AutoScale action.

add autoscale action <name> -type (SCALE_UP | SCALE_DOWN) -profileName <string> -parameters <string> [-vmDestroyGracePeriod <positive_integer>] [-quietTime <positive_integer>] -vServer <string>

name

ActionScale action name.

type

The type of action.

Possible values: SCALE_UP, SCALE_DOWN

profileName

AutoScale profile name.

parameters

Parameters to use in the action

vmDestroyGracePeriod

Time in minutes a VM is kept in inactive state before destroying

Default value: 10

quietTime

Time in seconds no other policy is evaluated or action is taken

Default value: 300

vServer

Name of the vserver on which autoscale action has to be taken.

Remove a AutoScale action.

rm autoscale action <name>

name

ActionScale action name.

Set a AutoScale action.

set autoscale action <name> [-profileName <string>] [-parameters <string>] [-vmDestroyGracePeriod <positive_integer>] [-quietTime <positive_integer>] [-vServer <string>]

name

ActionScale action name.

profileName

AutoScale profile name.

parameters

Parameters to use in the action

vmDestroyGracePeriod

Time in minutes a VM is kept in inactive state before destroying

Default value: 10

quietTime

Time in seconds no other policy is evaluated or action is taken

Default value: 300

vServer

Name of the vserver on which autoscale action has to be taken.

Use this command to remove autoscale action settings.Refer to the set autoscale action command for meanings of the arguments.

unset autoscale action <name> [-vmDestroyGracePeriod] [-quietTime]

Display the autoscale actions.

show autoscale action [<name>]

name

ActionScale action name.

summary

fullValues

format

level

type

The type of action.

profileName

AutoScale profile name.

parameters

Parameters to use in the action

vmDestroyGracePeriod

Time in minutes a VM is kept in inactive state before destroying

quietTime

Time in seconds no other policy is evaluated or action is taken

vServer

Name of the vserver on which autoscale action has to be taken.

destIP

IP Address on which provisioning server daemon is running

destPort

Port on which provisioning server daemon is running

devno

count

stateflag

autoscale policy

Sep 22, 2015

The following operations can be performed on "autoscale policy":

add | rm | set | unset | show | stat | rename

Create a autoscale policy.

add autoscale policy <name> -rule <expression> -action <string> [-comment <string>] [-logAction <string>]

name

The name of the autoscale policy.

rule

The rule associated with the policy.

action

The autoscale profile associated with the policy.

comment

Comments associated with this autoscale policy.

logAction

The log action associated with the autoscale policy

Remove a autoscale policy.

rm autoscale policy <name>

name

The name of the autoscale policy.

rm autoscale policy pol

Set a new rule/action/comment for an existing autoscale policy.

set autoscale policy <name> [-rule <expression>] [-action <string>] [-comment <string>] [-logAction <string>]

name

The name of the autoscale policy.

rule

The rule associated with the policy.

action

The autoscale profile associated with the policy.

comment

Comments associated with this autoscale policy.

logAction

The log action associated with the autoscale policy

set autoscaler policy pol -rule true

Unset comment/logaction for existing autoscale policy..Refer to the set autoscale policy command for meanings of the arguments.

unset autoscale policy <name> [-rule <expression>] [-action <string>] [-comment <string>] [-logAction <string>]

unset autoscale policy pol9 -undefAction

Display the autoscale policies.

show autoscale policy [<name>]

name

The name of the autoscale policy.

summary

fullValues

format

level

rule

The rule associated with the policy.

action

The autoscale profile associated with the policy.

comment

Comments associated with this autoscale policy.

logAction

The log action associated with the autoscale policy

stateflag

hits

Number of hits.

undefHits

Number of Undef hits.

priority

Specifies the priority of the policy.

boundTo

Location where policy is bound

activePolicy

Indicates whether policy is bound or not.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

devno

count

Display autoscale policy statistics.

stat autoscale policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

The name of the autoscale policy for which statistics will be displayed. If not given statistics are shown for all autoscale policies.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

stat autoscale policy

Rename a autoscale policy.

rename autoscale policy <name>@ <newName>@

name

The name of the autoscale policy.

newName

The new name of the autoscale policy.

rename autoscale policy oldname newname

autoscale profile

Sep 22, 2015

The following operations can be performed on "autoscale profile":

add | rm | set | show

Create a AutoScale policy.

add autoscale profile <name> -type CLOUDSTACK -url <URL> -apiKey -sharedSecret

name

AutoScale profile name.

type

The type of profile.

Possible values: CLOUDSTACK

url

URL providing the service

apiKey

api key for authentication with service

sharedSecret

shared secret for authentication with service

Remove a AutoScale policy.

rm autoscale profile <name>

name

AutoScale profile name.

Set a AutoScale policy.

set autoscale profile <name> [-url <URL>] [-apiKey] [-sharedSecret]

name

AutoScale profile name.

url

URL providing the service

apiKey

api key for authentication with service

sharedSecret

shared secret for authentication with service

Display the autoscale profile.

show autoscale profile [<name>]

name

AutoScale profile name.

summary

fullValues

format

level

type

The type of profile.

url

URL providing the service

apiKey

api key for authentication with service

sharedSecret

shared secret for authentication with service

stateflag

devno

count

Basic Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- configstatus
- dbsMonitors
- location
- locationData
- locationFile
- locationParameter
- nstrace
- reporting
- server
- service
- serviceGroup
- serviceGroupMember
- servicegroupbindings
- svcbindings
- uiinternal
- vserver

configstatus

Sep 22, 2015

The following operations can be performed on "configstatus":

Display status of packet engines.

show configstatus

consistent

State of packet engines.

culpritCore

Culprit core id.

core

Core id.

culpritCoreConfString

coreConfString

devno

count

stateflag

show configstatus

dbsMonitors

Sep 22, 2015

The following operations can be performed on "dbsMonitors":

Immediately send DNS queries to resolve the domain names of all the domain-based servers configured on the NetScaler appliance.

restart dbsMonitors

restart dbsMonitors

location

Sep 22, 2015

The following operations can be performed on "location":

add | rm | show

Creates a custom location entry on the NetScaler appliance. Custom locations can be used instead of a static location database if the number of locations you need does not exceed 500. Custom locations can also be used to override incorrect entries in the static database, because the appliance searches the static database before it searches the static location database.

add location <IPfrom> <IPto> <preferredLocation> [-longitude <integer> [-latitude <integer>]]

IPfrom

First IP address in the range, in dotted decimal notation.

IPto

Last IP address in the range, in dotted decimal notation.

preferredLocation

String of qualifiers, in dotted notation, describing the geographical location of the IP address range. Each qualifier is more specific than the one that precedes it, as in continent.country.region.city.isp.organization. For example, "NA.US.CA.San Jose.ATT.citrix".

Note: A qualifier that includes a dot (.) or space ( ) must be enclosed in double quotation marks.

longitude

Numerical value, in degrees, specifying the longitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

Minimum value: -180

Maximum value: 180

latitude

Numerical value, in degrees, specifying the latitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

Minimum value: -90

Maximum value: 90

Add location 192.168.100.1 192.168.100.100 *.us.ca.san jose

Removes a custom location entry from the NetScaler appliance.

rm location <IPfrom> <IPto>

IPfrom

First IP address in the range, in dotted decimal notation.

IPto

Last IP address in the range, in dotted decimal notation.

rm location 192.168.100.1 192.168.100.100

Displays all the custom location entries configured on the NetScaler appliance, or just the entry for the specified IP address range.

show location [<IPfrom>]

IPfrom

The qualifiers in dotted notation for the ipaddress. If this value is not specified, all custom entries are displayed.

summary

fullValues

format

level

IPto

The end of the IP address range.

preferredLocation

The qualifiers in dotted notation for the ipaddress range.

q1label

Least specific location qualifier.

q2label

Location qualifier 2.

q3label

Location qualifier 3.

q4label

Location qualifier 4.

q5label

Location qualifier 5.

q6label

Most specific location qualifier.

longitude

Numerical value, in degrees, specifying the longitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

latitude

Numerical value, in degrees, specifying the latitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

devno

count

stateflag

show location

locationData

Sep 22, 2015

The following operations can be performed on "locationData":

Clears all location information, including custom and static database entries.

clear locationData

clear locationdata

locationFile

Sep 22, 2015

The following operations can be performed on "locationFile":

add | rm | show

Loads the static location database from the specified file.

add locationFile <locationFile> [-format <format>]

locationFile

Name of the location file, with or without absolute path. If the path is not included, the default path (/var/netscaler/locdb) is assumed. In a high availability setup, the static database must be stored in the same location on both NetScaler appliances.

format

Format of the location file. Required for the NetScaler appliance to identify how to read the location file.

Possible values: netscaler, ip-country, ip-country-isp, ip-country-region-city, ip-country-region-city-isp, geoip-country, geoip-region, geoip-city, geoip-country-org, geoip-country-isp, geoip-city-isp-org

Default value: NSMAP_FORMAT_NETSCALER

```
add locationfile /var/nsmap/locationdb  -format netscaler
```

Removes the currently loaded static location database from the NetScaler appliance.

rm locationFile

```
rm locationfile
```

Displays the name, including the absolute path, and format of the location file currently loaded on the NetScaler appliance.

show locationFile

format

level

locationFile

The name of the location file.

format

The format of the location file.

show locationfile

locationParameter

Sep 22, 2015

The following operations can be performed on "locationParameter":

set | unset | show

Sets the location parameters used for static-proximity based global server load balancing. Location parameters include up to six qualifiers and a context that specifies how the qualifiers must be interpreted. Each qualifier specifies the location of an IP address range and is more specific than the one that precedes it, as in continent.country.region.city.isp.organization. For example, "NA.US.CA.San Jose.ATT.citrix". Note: A qualifier that includes a dot (.) or space ( ) must be enclosed in double quotation marks.

set locationParameter [-context (geographic | custom)] [-q1label <string>] [-q2label <string>] [-q3label <string>] [-q4label <string>] [-q5label <string>] [-q6label <string>]

context

Context for describing locations. In geographic context, qualifier labels are assigned by default in the following sequence: Continent.Country.Region.City.ISP.Organization. In custom context, the qualifiers labels can have any meaning that you designate.

Possible values: geographic, custom

q1label

Label specifying the meaning of the first qualifier. Can be specified for custom context only.

q2label

Label specifying the meaning of the second qualifier. Can be specified for custom context only.

q3label

Label specifying the meaning of the third qualifier. Can be specified for custom context only.

q4label

Label specifying the meaning of the fourth qualifier. Can be specified for custom context only.

q5label

Label specifying the meaning of the fifth qualifier. Can be specified for custom context only.

q6label

Label specifying the meaning of the sixth qualifier. Can be specified for custom context only.

set locationparameter -context custom

Use this command to remove locationParameter settings.Refer to the set locationParameter command for meanings of the arguments.

unset locationParameter [-context] [-q1label] [-q2label] [-q3label] [-q4label] [-q5label] [-q6label]

Displays current values for the location parameters, which are used for static-proximity based load balancing.

show locationParameter

format

level

context

The context in which a static proximity decision must be made.

q1label

The label for the 1st qualifier. These qualifier labels characterize the locations mapped with the IP addresses used to make static proximity decisions.

q2label

Label specifying the meaning of the second qualifier. Can be specified for custom context only.

q3label

Label specifying the meaning of the third qualifier. Can be specified for custom context only.

q4label

Label specifying the meaning of the fourth qualifier. Can be specified for custom context only.

q5label

Label specifying the meaning of the fifth qualifier. Can be specified for custom context only.

q6label

Label specifying the meaning of the sixth qualifier. Can be specified for custom context only.

locationFile

Currently loaded location database file.

format

custom

Number of configured custom locations.

static

Number of configured locations in the database file (static locations).

lines

Number of lines in the databse files

errors

Number of errros encountered while reading the database file.

warnings

Number of warnings encountered while reading the database file.

entries

Number of successfully added entries.

flags

Information needed for display. This argument passes information from the kernel to the user space.

status

This argument displays when the status (success or failure) of database loading.

DatabaseMode

This argument displays the database mode.

flushing

This argument displays the state of flushing.

loading

This argument displays the state of loading.

```
show locationparameter
```

nstrace

Sep 22, 2015

The following operations can be performed on "nstrace":

start | stop | show

Start NetScaler packet capture tool.

start nstrace [-nf <positive_integer>] [-time <positive_integer>] [-size <positive_integer>] [-mode <mode> ...] [-tcpdump (ENABLED | DISABLED)] [-perNIC (ENABLED | DISABLED)] [-fileName <string>] [-fileId <string>] [-filter <expression>] [-link (ENABLED | DISABLED)] [-nodes <positive_integer> ...]

nf

Number of files to be generated in cycle.

Default value: 24

Minimum value: 1

Maximum value: 100

time

Time per file (sec).

Default value: 3600

Minimum value: 1

size

Size of the captured data. Set 0 for full packet trace.

Default value: 164

Maximum value: 1514

mode

Capturing mode for trace. Mode can be any of the following values or combination of these values:

RX Received packets before NIC pipelining

NEW_RX Received packets after NIC pipelining

TX Transmitted packets

TXB Packets buffered for transmission

IPV6 Translated IPv6 packets

C2C Capture C2C message

NS_FR_TX TX/TXB packets are not captured in flow receiver.

Default mode: NEW_RX TXB

Default value: DEFAULT_MODE

tcpdump

Trace is captured in TCPDUMP(.pcap) format. Default capture format is NSTRACE(.cap).

Possible values: ENABLED, DISABLED

Default value: DISABLED

perNIC

Use separate trace files for each interface. Works only with tcpdump format.

Possible values: ENABLED, DISABLED

Default value: DISABLED

fileName

Name of the trace file.

fileId

ID for the trace file name for uniqueness. Should be used only with -name option.

filter

Filter expression for nstrace. Maximum length of filter is 255 and it can be of following format:

<expression> [<relop> <expression>]

<relop> = (&& | ||)

nstrace supports two types of filter expressions:

Classic Expressions:

<expression> = the expression string in the format:

<qualifier> <operator> <qualifier-value>

<qualifier> = SOURCEIP.

<qualifier-value> = A valid IP address

<qualifier> = SOURCEPORT.

<qualifier-value> = A valid port number.

<qualifier> = DESTIP.

<qualifier-value> = A valid IP address.

<qualifier> = DESTPORT.

<qualifier-value> = A valid port number.

<qualifier> = IP.

<qualifier-value> = A valid IP address.

<qualifier> = PORT.

<qualifier-value> = A valid port number.

<qualifier> = SVCNAME.

<qualifier-value> = The name of a service.

<qualifier> = VSVRNAME.

<qualifier-value> = The name of a vserver.

<qualifier> = CONNID

<qualifier-value> = A valid PCB dev number.

<qualifier> = VLAN

<qualifier-value> = A valid VLAN ID.

<qualifier> = INTF

<qualifier-value> = A valid interface id in the form of x/y

(n/x/y in case of cluster interface).

<operator> = (== | eq | != | neq | > | gt

| < | lt | >= | ge | <= | le | BETWEEN)

eg: start nstrace -filter "SOURCEIP == 10.102.34.201 || (SVCNAME != s1 && SOURCEPORT > 80)"

The filter expression should be given in double quotes.

Default Expressions:

<expression> =:

CONNECTION.<qualifier>.<qualifier-method>.(<qualifier-value>)

<qualifier> = SRCIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.SRCIP.EQ(127.0.0.1)

<qualifier> = DSTIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.DSTIP.EQ(127.0.0.1)

<qualifier> = IP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.IP.EQ(127.0.0.1)

<qualifier> = SRCIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.SRCIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = DSTIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.DSTIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = IPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.IPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = SRCPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid port number.

example = CONNECTION.SRCPORT.EQ(80)

<qualifier> = DSTPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid port number.

example = CONNECTION.DSTPORT.EQ(80)

<qualifier> = PORT

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid port number.

example = CONNECTION.PORT.EQ(80)

<qualifier> = VLANID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid vlan ID.

example = CONNECTION.VLANID.EQ(0)

<qualifier> = CONNID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid PCB dev number.

example = CONNECTION.CONNID.EQ(0)

<qualifier> = PPEID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid core ID.

example = CONNECTION.PPEID.EQ(0)

<qualifier> = SVCNAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH

| ENDSWITH]

<qualifier-value> = A valid text string.

example = CONNECTION.SVCNAME.EQ("name")

<qualifier> = INTF

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid interface id in the

form of x/y.

example = CONNECTION.INTF.EQ("x/y")

eg: start nstrace -filter "CONNECTION.SRCIP.EQ(127.0.0.1) || (CONNECTION.SVCNAME.NE("s1") && CONNECTION.SRCPORT.EQ(80))"

The filter expression should be given in double quotes.

common use cases:

Trace capturing full sized traffic from/to ip 10.102.44.111, excluding loopback traffic

start nstrace -size 0 -filter "CONNECTION.IP.NE(127.0.0.1) && CONNECTION.IP.EQ(10.102.44.111)"

Trace capturing all traffic to (terminating at) port 80 or 443

start nstrace -size 0 -filter "CONNECTION.DSTPORT.EQ(443) || CONNECTION.DSTPORT.EQ(80)"

Trace capturing all backend traffic specific to service service1 along with corresponding client side traffic

start nstrace -size 0 -filter "CONNECTION.SVCNAME.EQ("service1")" -link ENABLED

Trace capturing all traffic through NS interface 1/1

start nstrace -filter "CONNECTION.INTF.EQ("1/1")"

Trace capturing all traffic specific through vlan 2

start nstrace -filter "CONNECTION.VLANID.EQ(2)"

Trace capturing all frontend (client side) traffic specific to lb vserver vserver1 along with corresponding server side traffic

start nstrace -size 0 -filter "CONNECTION.LB_VSERVER.NAME.EQ("vserver1")" -link ENABLED

link

Includes filtered connection's peer traffic.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nodes

Nodes on which tracing is started.

Maximum value: 32

start nstrace -time 10

Stop running NetScaler packet capture tool.

stop nstrace

stop nstrace

Display nstrace parameters set through 'start nstrace' command.

show nstrace

state

Current running state of trace.

scope

Scope of started trace, local or cluster level.

traceLocation

Directory where current trace files are saved.

nf

Number of files to be generated in cycle.

time

Time per file (sec).

size

Size of the captured data. Set 0 for full packet trace.

mode

Capturing mode for trace. Mode can be any of the following values or combination of these values:

RX Received packets before NIC pipelining

NEW_RX Received packets after NIC pipelining

TX Transmitted packets

TXB Packets buffered for transmission

IPV6 Translated IPv6 packets

C2C Capture C2C message

NS_FR_TX TX/TXB packets are not captured in flow receiver.

Default mode: NEW_RX TXB

tcpdump

Trace is captured in TCPDUMP(.pcap) format. Default capture format is NSTRACE(.cap).

perNIC

Use separate trace files for each interface. Works only with tcpdump format.

fileName

Name of the trace file.

fileId

ID for the trace file name for uniqueness. Should be used only with -name option.

filter

Filter expression for nstrace. Maximum length of filter is 255 and it can be of following format:

<expression> [<relop> <expression>]

<relop> = (&& | ||)

nstrace supports two types of filter expressions:

Classic Expressions:

<expression> = the expression string in the format:

<qualifier> <operator> <qualifier-value>

<qualifier> = SOURCEIP.

<qualifier-value> = A valid IP address

<qualifier> = SOURCEPORT.

<qualifier-value> = A valid port number.

<qualifier> = DESTIP.

<qualifier-value> = A valid IP address.

<qualifier> = DESTPORT.

<qualifier-value> = A valid port number.

<qualifier> = IP.

<qualifier-value> = A valid IP address.

<qualifier> = PORT.

<qualifier-value> = A valid port number.

<qualifier> = SVCNAME.

<qualifier-value> = The name of a service.

<qualifier> = VSVRNAME.

<qualifier-value> = The name of a vserver.

<qualifier> = CONNID

<qualifier-value> = A valid PCB dev number.

<qualifier> = VLAN

<qualifier-value> = A valid VLAN ID.

<qualifier> = INTF

<qualifier-value> = A valid interface id in the form of x/y

(n/x/y in case of cluster interface).

<operator> = (== | eq | != | neq | > | gt

| < | lt | >= | ge | <= | le | BETWEEN)

eg: start nstrace -filter "SOURCEIP == 10.102.34.201 || (SVCNAME != s1 && SOURCEPORT > 80)"

The filter expression should be given in double quotes.

Default Expressions:

<expression> =:

CONNECTION.<qualifier>.<qualifier-method>.(<qualifier-value>)

<qualifier> = SRCIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.SRCIP.EQ(127.0.0.1)

<qualifier> = DSTIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.DSTIP.EQ(127.0.0.1)

<qualifier> = IP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.IP.EQ(127.0.0.1)

<qualifier> = SRCIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.SRCIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = DSTIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.DSTIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = IPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.IPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = SRCPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid port number.

example = CONNECTION.SRCPORT.EQ(80)

<qualifier> = DSTPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid port number.

example = CONNECTION.DSTPORT.EQ(80)

<qualifier> = PORT

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid port number.

example = CONNECTION.PORT.EQ(80)

<qualifier> = VLANID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid vlan ID.

example = CONNECTION.VLANID.EQ(0)

<qualifier> = CONNID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid PCB dev number.

example = CONNECTION.CONNID.EQ(0)

<qualifier> = PPEID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE]

<qualifier-value> = A valid core ID.

example = CONNECTION.PPEID.EQ(0)

<qualifier> = SVCNAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH

| ENDSWITH]

<qualifier-value> = A valid text string.

example = CONNECTION.SVCNAME.EQ("name")

<qualifier> = INTF

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid interface id in the

form of x/y.

example = CONNECTION.INTF.EQ("x/y")

eg: start nstrace -filter "CONNECTION.SRCIP.EQ(127.0.0.1) || (CONNECTION.SVCNAME.NE("s1") && CONNECTION.SRCPORT.EQ(80))"

The filter expression should be given in double quotes.

common use cases:

Trace capturing full sized traffic from/to ip 10.102.44.111, excluding loopback traffic

start nstrace -size 0 -filter "CONNECTION.IP.NE(127.0.0.1) && CONNECTION.IP.EQ(10.102.44.111)"

Trace capturing all traffic to (terminating at) port 80 or 443

start nstrace -size 0 -filter "CONNECTION.DSTPORT.EQ(443) || CONNECTION.DSTPORT.EQ(80)"

Trace capturing all backend traffic specific to service service1 along with corresponding client side traffic

start nstrace -size 0 -filter "CONNECTION.SVCNAME.EQ("service1")" -link ENABLED

Trace capturing all traffic through NS interface 1/1

start nstrace -filter "CONNECTION.INTF.EQ("1/1")"

Trace capturing all traffic specific through vlan 2

start nstrace -filter "CONNECTION.VLANID.EQ(2)"

Trace capturing all frontend (client side) traffic specific to lb vserver vserver1 along with corresponding server side traffic

start nstrace -size 0 -filter "CONNECTION.LB_VSERVER.NAME.EQ("vserver1")" -link ENABLED

link

Includes filtered connection's peer traffic.

nodes

Nodes on which tracing is started.

show nstrace

reporting

Sep 22, 2015

The following operations can be performed on "reporting":

enable | disable | show

Enable the data collection for reporting module.

enable reporting

enable reporting

Disable the data collection for reporting module.

disable reporting

disable reporting

---

show the state of data collection for reporting module.

show reporting

state

The rule associated with the entity

show reporting

---

server

Sep 22, 2015

The following operations can be performed on "server":

add | rm | set | unset | enable | disable | show | rename

Creates a server entry on the NetScaler appliance. The NetScaler appliance supports two types of servers: IP address based servers and domain based servers.

add server <name>@ (<IPAddress>@ | (<domain>@ [-domainResolveRetry <integer>] [-IPv6Address (YES | NO)]) | (-translationIp <ip_addr> -translationMask <netmask>)) [-state (ENABLED | DISABLED)] [-comment <string>] [-td <positive_integer>]

name

Name for the server.

Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Can be changed after the name is created.

IPAddress

IPv4 or IPv6 address of the server. If you create an IP address based server, you can specify the name of the server, instead of its IP address, when creating a service. Note: If you do not create a server entry, the server IP address that you enter when you create a service becomes the name of the server.

domain

Domain name of the server. For a domain based configuration, you must create the server first.

translationIp

IP address used to transform the server's DNS-resolved IP address.

domainResolveRetry

Time, in seconds, for which the NetScaler appliance must wait, after DNS resolution fails, before sending the next DNS query to resolve the domain name.

Default value: 5

Minimum value: 5

Maximum value: 20939

state

Initial state of the server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

IPv6Address

Support IPv6 addressing mode. If you configure a server with the IPv6 addressing mode, you cannot use the server in the IPv4 addressing mode.

Possible values: YES, NO

Default value: NO

comment

Any information about the server.

td

Traffic Domain Id.

Maximum value: 4094

add server web_serv 10.102.27.150 To add multiple servers you can use the following command: add server serv[1-3] 10.102.27.[151-153] The above command adds thre

---

Removes a server entry from the NetScaler appliance.

rm server <name>@ ...

name

Name of the server entry to remove.

rm server web_svr To remove the servers named serv1, serv2 and serv3 at once you can use the following command: rm server serv[1-3]

Modifies the specified parameters of a server entry.

set server <name>@ [-IPAddress <ip_addr|ipv6_addr|*>@ | -domainResolveRetry <integer> | -translationIp <ip_addr> | -translationMask <netmask> | -domainResolveNow] [-comment <string>]

name

Name of the server whose parameters you are configuring.

IPAddress

Name of the server whose parameters you are configuring.

domainResolveRetry

Time, in seconds, for which the NetScaler appliance must wait, after DNS resolution fails, before sending the next DNS query to resolve the domain name.

Default value: 5

Minimum value: 5

Maximum value: 20939

translationIp

IP address used to transform the server's DNS-resolved IP address.

translationMask

The netmask of the translation ip

domainResolveNow

Immediately send a DNS query to resolve the server's domain name.

comment

Any information about the server.

set server http_svr -IPAddress 10.102.1.112 To set multiple servers IP addresses at onece you can use the following command: setserver serv[1-3] -IPAddress 10.102.27.

Use this command to remove server settings.Refer to the set server command for meanings of the arguments.

unset server <name>@ -comment

Enables all services on the specified server.

enable server <name>@

name

Name of the server to enable.

enable server web_serv To enable all the services configured on servers named serv1, serv2 and serv3 at once, use the following command: enable server serv[1-3]

Disables all services on the server. When a server is disabled, all services on the server are disabled.

disable server <name>@ [<delay>] [-graceFul (YES | NO)]

name

Name of the server to disable.

delay

Time, in seconds, after which all the services configured on the server are disabled.

graceFul

Shut down gracefully, without accepting any new connections, and disabling each service when all of its connections are closed.

Possible values: YES, NO

Default value: NO

disable server web_svr 30 To disable all the services configured on servers named serv1, serv2 and serv3 at once, use the following command: disable server serv[1-3]

Displays the parameters of all the server entries on the appliance, or the parameters of the specified server entry.

show server [<name> | -internal]

name

Name of the server for which to display parameters.

internal

Display names of the servers that have been created for internal use.

summary

fullValues

format

level

IPAddress

The IP Address of server.

state

The State of the server.

domain

The domain name of the server.

domainResolveRetry

Time, in seconds, for which the NetScaler appliance must wait, after DNS resolution fails, before sending the next DNS query to resolve the domain name.

serviceName

The services attatched to the server.

serviceGroupName

servicegroups bind to this server

translationIp

IP address used to transform the server's DNS-resolved IP address.

translationMask

The netmask of the translation ip

comment

Any information about the server.

stateflag

stateflag

serviceType

service type of the service.

serviceIPAddress

The IP address of the bound service

serviceIPstr

This field has been intorduced to show the dbs services ip

port

port of the service.

svrState

The state of the bound service

stateChangeTimeSec

Time when last state change happened. Seconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

IPv6Address

Support IPv6 addressing mode. If you configure a server with the IPv6 addressing mode, you cannot use the server in the IPv4 addressing mode.

svrcfgFlags

service flags to denote its a db enabled.

td

Traffic Domain Id.

autoScale

Auto scale option for a servicegroup

devno

count

> show server web_svr1 Name: web_svr1 State:ENABLED IPAddress: 10.102.27.154 > show server web_svr1 Name: web

---

Renames a server.

rename server <name>@ <newName>@

name

Existing name of the server.

newName

New name for the server. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

rename server s1 s1-new

---

service

Sep 22, 2015

The following operations can be performed on "service":

add | rm | set | unset | bind | unbind | enable | disable | show | rename | stat

Creates a service on the NetScaler appliance. If the service is domain based, before you create the service, create the server entry by using the add server command. Then, in this command, specify the Server parameter.

add service <name>@ (<IP>@ | <serverName>@) <serviceType> <port> [-clearTextPort <port>] [-cacheType <cacheType>] [-maxClient <positive_integer>] [-healthMonitor (YES | NO)] [-maxReq <positive_integer>] [-cacheable (YES | NO)] [-cip (ENABLED | DISABLED) [<cipHeader>]] [-usip (YES | NO)] [-pathMonitor (YES | NO)] [-pathMonitorIndv (YES | NO)] [-useproxyport (YES | NO)] [-sc (ON | OFF)] [-sp (ON | OFF)] [-rtspSessionidRemap (ON | OFF)] [-cltTimeout <secs>] [-svrTimeout <secs>] [-CustomServerID <string>] [-CKA (YES | NO)] [-TCPB (YES | NO)] [-CMP (YES | NO)] [-maxBandwidth <positive_integer>] [-accessDown (YES | NO)] [-monThreshold <positive_integer>] [-state (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-tcpProfileName <string>] [-httpProfileName <string>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-td <positive_integer>]

name

Name for the service. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the service has been created.

IP

IP to assign to the service.

serverName

Name of the server that hosts the service.

serviceType

Protocol in which data is exchanged with the service.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, RPCSVR, DNS, ADNS, SNMP, RTSP, DHCPRA, ANY, SIP_UDP, DNS_TCP, ADNS_TCP, MYSQL, MSSQL, RADIUS, RDP, DIAMETER, SSL_DIAMETER, TFTP

port

Port number of the service.

clearTextPort

Port to which clear text data must be sent after the appliance decrypts incoming SSL traffic. Applicable to transparent SSL services.

Minimum value: 1

cacheType

Cache type supported by the cache server.

Possible values: TRANSPARENT, REVERSE, FORWARD

maxClient

Maximum number of simultaneous open connections to the service.

Maximum value: 4294967294

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

maxReq

Maximum number of requests that can be sent on a persistent connection to the service.

Note: Connection requests beyond this value are rejected.

Maximum value: 65535

cacheable

Use the transparent cache redirection virtual server to forward requests to the cache server.

Note: Do not specify this parameter if you set the Cache Type parameter.

Possible values: YES, NO

Default value: NO

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

Possible values: ENABLED, DISABLED

cipHeader

Name for the HTTP header whose value must be set to the IP address of the client. Used with the Client IP parameter. If you set the Client IP parameter, and you do not specify a name for the header, the appliance uses the header name specified for the global Client IP Header parameter (the cipHeader parameter in the set ns param CLI command or the Client IP Header parameter in the Configure HTTP Parameters dialog box at System > Settings > Change HTTP parameters). If the global Client IP Header parameter is not specified, the appliance inserts a header with the name "client-ip."

usip

Use the client's IP address as the source IP address when initiating a connection to the server. When creating a service, if you do not set this parameter, the service inherits the global Use Source IP setting (available in the enable ns mode and disable ns mode CLI commands, or in the System > Settings > Configure modes > Configure Modes dialog box). However, you can override this setting after you create the service.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

sc

State of SureConnect for the service.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service.

Possible values: ON, OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service.

Possible values: ON, OFF

Default value: OFF

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CustomServerID

Unique identifier for the service. Used when the persistency type for the virtual server is set to Custom Server ID.

Default value: "None"

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.

CKA

Enable client keep-alive for the service.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service.

Possible values: YES, NO

CMP

Enable compression for the service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated to the service.

Maximum value: 4294967287

accessDown

Use Layer 2 mode to bridge the packets sent to this service if it is marked as DOWN. If the service is DOWN, and this parameter is disabled, the packets are dropped.

Possible values: YES, NO

Default value: NO

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Maximum value: 65535

state

Initial state of the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

downStateFlush

Perform delayed clean-up of connections to the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service.

hashId

A numerical identifier that can be used by hash based load balancing methods. Must be unique for each service.

Minimum value: 1

comment

Any information about the service.

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile to use for the service.

td

Traffic Domain Id.

Maximum value: 4094

add service http_svc 10.102.1.112 http 80 The below command adds the service web_svc1 for the server web_serv1, web_svc2 for web_serv2 and web_svc3 for web_serv3.

Removes a service.

rm service <name>@

name

Name of the service.

rm service http_svc To remove services svc1, svc2 and svc3 in one go use the following command: rm service svc[1-3]

Modifies the parameters of an existing service.

set service <name>@ [-IPAddress <ip_addr|ipv6_addr|*>@] [-maxClient <positive_integer>] [-maxReq <positive_integer>] [-cacheable (YES | NO)] [-cip (ENABLED | DISABLED) [<cipHeader>]] [-usip (YES | NO)] [-pathMonitor (YES | NO)] [-pathMonitorIndv (YES | NO)] [-useproxyport (YES | NO)] [-sc (ON | OFF)] [-sp (ON | OFF)] [-rtspSessionidRemap (ON | OFF)] [-healthMonitor (YES | NO)] [-cltTimeout <secs>] [-svrTimeout <secs>] [-CustomServerID <string>] [-CKA (YES | NO)] [-TCPB (YES | NO)] [-CMP (YES | NO)] [-maxBandwidth <positive_integer>] [-accessDown (YES | NO)] [-monThreshold <positive_integer>] [-weight <positive_integer> <monitorName>] [-downStateFlush (ENABLED | DISABLED)] [-tcpProfileName <string>] [-httpProfileName <string>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>]

name

Name of the service for which to modify parameters.

IPAddress

The new IP address of the service.

maxClient

Maximum number of simultaneous open connections to the service.

Maximum value: 4294967294

maxReq

Maximum number of requests that can be sent on a persistent connection to the service.

Note: Connection requests beyond this value are rejected.

Maximum value: 65535

cacheable

Use the transparent cache redirection virtual server to forward requests to the cache server.

Note: Do not specify this parameter if you set the Cache Type parameter.

Possible values: YES, NO

Default value: NO

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

Possible values: ENABLED, DISABLED

usip

Use the client's IP address as the source IP address when initiating a connection to the server. When creating a service, if you do not set this parameter, the service inherits the global Use Source IP setting (available in the enable ns mode and disable ns mode CLI commands, or in the System > Settings > Configure modes > Configure Modes dialog box). However, you can override this setting after you create the service.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

sc

State of SureConnect for the service.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service.

Possible values: ON, OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service.

Possible values: ON, OFF

Default value: OFF

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CustomServerID

Unique identifier for the service. Used when the persistency type for the virtual server is set to Custom Server ID.

Default value: "None"

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.

CKA

Enable client keep-alive for the service.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service.

Possible values: YES, NO

CMP

Enable compression for the service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated to the service.

Maximum value: 4294967287

accessDown

Use Layer 2 mode to bridge the packets sent to this service if it is marked as DOWN. If the service is DOWN, and this parameter is disabled, the packets are dropped.

Possible values: YES, NO

Default value: NO

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Maximum value: 65535

weight

Weight to assign to the monitor-service binding. When a monitor is UP, the weight assigned to its binding with the service determines how much the monitor contributes toward keeping the health of the service above the value configured for the Monitor Threshold parameter.

Minimum value: 1

Maximum value: 100

downStateFlush

Perform delayed clean-up of connections to the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service.

hashId

A numerical identifier that can be used by hash based load balancing methods. Must be unique for each service.

Minimum value: 1

comment

Any information about the service.

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile to use for the service.

set service http_svc -maxClient 100 The following command sets IP address 10.102.27.53 for service svc1, 10.102.27.54 for svc2 and 10.102.27.55 for svc3. set service :

Removes the parameter settings of the specified service. Attributes for which a default value is available revert to their default values..Refer to the set service command for meanings of the arguments.

unset service <name>@ [-maxClient] [-maxReq] [-cacheable] [-cip] [-usip] [-pathMonitor] [-pathMonitorIndv] [-useproxyport] [-sc] [-sp] [-rtspSessionidRemap] [-CustomServerID] [-CKA] [-TCPB] [-CMP] [-maxBandwidth] [-accessDown] [-monThreshold] [-cltTimeout] [-svrTimeout] [-tcpProfileName] [-httpProfileName] [-hashId] [-appflowLog] [-netProfile] [-cipHeader] [-healthMonitor] [-monitorName] [-downStateFlush] [-comment]

unset service http_svc -maxClient To unset maxclients for services svc1, svc2 and svc3, the following command can be used: unset service svc[1-3] -maxClient

---

Binds a policy or a monitor to a service.

bind service <name>@ (-policyName <string> | (-monitorName <string>@ [-monState (ENABLED | DISABLED)] [-weight <positive_integer>] [-passive]))

name

Name of the service to which to bind a policy or monitor.

policyName

Name of the policy to bind to the service.

monitorName

Name of the monitor to bind to the service.

bind service svc1 -policyName pol1 To bind svc1, svc2 and svc3 to the policy pol1 you can use the following command: bind service svc[1-3] -policyName pol1

Unbinds a policy or monitor from the specified service.

unbind service <name>@ (-policyName <string> | -monitorName <string>@)

name

Name of the service from which to unbind a policy or monitor.

policyName

Name of the policy to unbind.

monitorName

Name of the monitor assigned to the service.

unbind service http_svc -policyName pol1 To unbind a policy called pol1 on services svc1, svc2 and svc3, use the following command: unbind service svc[1-3] -policyNam

---

Enables a service.

enable service <name>@

name

Name of the service.

enable service http_svc To enable svc1, svc2 and svc3 in one go use the following command: enable service svc[1-3]

Disables a service.

disable service <name>@ [<delay>] [-graceFul (YES | NO)]

name

Name of the service.

delay

Time, in seconds, allocated to the NetScaler appliance for a graceful shutdown of the service. During this period, new requests are sent to the service only for clients who already have persistent sessions on the appliance. Requests from new clients are load balanced among other available services. After the delay time expires, no requests are sent to the service, and the service is marked as unavailable (OUT OF SERVICE).

graceFul

Shut down gracefully, not accepting any new connections, and disabling the service when all of its connections are closed.

Possible values: YES, NO

Default value: NO

disable service http_svc 10 To disable svc1, svc2 and svc3 in one go use the following command: disable service svc[1-3] 10

Displays a list of all services configured on the NetScaler appliance, or the configuration details of the specified service.

show service [<name> | -all | -internal] show service bindings - alias for 'show svcbindings'

name

Name of the service for which to display configuration details.

all

Display both user-configured and dynamically learned services.

internal

Display only dynamically learned services.

summary

fullValues

format

level

numOfconnections

This will tell the number of client side connections are still open.

serverName

The name of the server for which a service has created.

policyName

The name of the policyname for which this service is bound

serviceType

The type of service

serviceConfTpye

The configuration type of the serviceNOTE: This attribute is deprecated.This will no longer show the correct information. Use the serviceConfType option instead.

serviceConfType

The configuration type of the serviceNOTE: This attribute is deprecated.This will no longer show the correct information. Use the serviceConfType2 option instead.

serviceConfType2

The configuration type of the service (Internal/Dynamic/Configured).

port

Port number of the service.

value

SSL status.

clearTextPort

The clear-text port number where clear-text data is sent. Used with SSL offload service

gslb

The GSLB option for the corresponding virtual server.

cacheType

Cache type supported by the cache server.

maxClient

Maximum number of simultaneous open connections to the service.

maxReq

Maximum number of requests that can be sent on a persistent connection to the service.

Note: Connection requests beyond this value are rejected.

cacheable

Use the transparent cache redirection virtual server to forward requests to the cache server.

Note: Do not specify this parameter if you set the Cache Type parameter.

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

cipHeader

The client IP header.

usip

The use of client's IP Address option.

pathMonitor

Path monitoring for clustering

pathMonitorIndv

Individual Path monitoring for decisions.

useproxyport

The use of client's Port.

sc

The state of SureConnect for the service.

weight

The weight for the specified monitor.

state

Initial state of the service.

sp

Enable surge protection for the service.

rtspSessionidRemap

Enable RTSP session ID mapping for the service.

failedprobes

Number of the current failed monitoring probes.

cltTimeout

The idle time in seconds after which the client connection is terminated.

totalprobes

The total number of probs sent.

svrTimeout

The idle time in seconds after which the server connection is terminated.

totalfailedprobes

The total number of failed probs.

publicIP

public ip

publicPort

public port

CustomServerID

The identifier for the service. Used when the persistency type is set to Custom Server ID.

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.NOTE: This attribute is deprecated.Instead of integer now serverId will be a string and you can use -customserverid instead of -serverID.

CKA

Enable client keep-alive for the service.

TCPB

Enable TCP buffering for the service.

CMP

Enable compression for the service.

maxBandwidth

The maximum bandwidth in kbps allowed for the service

accessDown

The option to allow access to disabled or down services. If enabled, all packets to the service are bridged; if disabled, they are dropped.

svrState

The state of the service

delay

The remaining time in seconds for the service to be disabled

IPAddress

The IP address of the server.

monitorName

The monitor Names.

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

monState

The running state of the monitor on this service.

monStatCode

The code indicating the monitor response.

lastresponse

The string form of monstatcode.

responseTime

Response time of this monitor.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

downStateFlush

Perform delayed clean-up of connections to the service.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimemSec

Time at which last state change happened. Milliseconds part.

timeSinceLastStateChange

Time in milliseconds since the last state change.NOTE: This attribute is deprecated.This will no longer show the correct information. Use the ticksSinceLastStateChange option instead.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

StateUpdateReason

Checks state update reason on the secondary node.

ClMonOwner

Tells the mon owner of the service.

ClMonView

Tells the view id of the monitoring owner.

tcpProfileName

Name of the TCP profile.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service.

hashId

A numerical identifier that can be used by hash based load balancing methods. Must be unique for each service.

graceFul

Indicates graceful shutdown of the service. System will wait for all outstanding connections to this service to be closed before disabling the service.

comment

Comments associated with this service.

monitorTotalProbes

Total number of probes sent to monitor this service.

monitorTotalFailedProbes

Total number of failed probes

monitorCurrentFailedProbes

Total number of currently failed probes

stateflag

stateflag

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

appflowLog

Enable logging of AppFlow information.

netProfile

Network profile to use for the service.

svccfgFlags

Contains the information about config info like internal/configured service

serviceIPstr

This field has been intorduced to show the dbs services ip

svcMonFlags

to store the flags of monitor bound to it

td

Traffic Domain Id.

passive

Indicates if load monitor is passive. A passive load monitor does not remove service from LB decision when threshold is breached.

devno

count

The following is sample output of the show service -all command: 4 configured services: 1) svc1 (10.124.99.12:80) - HTTP State: UP

Renames a service.

rename service <name>@ <newName>@

name

Existing name of the service to be renamed.

newName

New name for the service. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

rename service svc1 svcnew

Displays statistics that have been collected for the specified service.

stat service [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the service.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Throughput (Mbps) (Throughput)

Number of bytes received or sent by this service (Mbps).

Average server TTFB (SvrTTFB)

Average TTFB between the NetScaler appliance and the server.TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Service type (Type)

The service type of this service.Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Current client connections (ClntConn)

Number of current client connections.

Requests in surge queue (SurgeQ)

Number of requests in the surge queue.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Connections in reuse pool (ReuseP)

Number of requests in the idle queue/reuse pool.

Maximum server connections (MaxConn)

Maximum open connections allowed on this service.

Current load on the service (Load)

Load on the service that is calculated from the bound load based monitor.

Current flags on the service (CurtFlags)

Current flags on the service for internal use in display handlers.

Service hits (Hits)

Number of times that the service has been provided.

ActvTrans

Number of active transactions handled by this service. (Including those in the surge queue.)

Active Transaction means number of transactions currently served by the server including those waiting in the SurgeQ

Total Packets rcvd (PktRx)

Total number of packets received by this service or virtual server.

Total Packets sent (PktTx)

Total number of packets sent.

serviceGroup

Sep 22, 2015

The following operations can be performed on "serviceGroup":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

Creates a service group. You can group similar services into a service group and use them as a single entity.

add serviceGroup <serviceGroupName>@ <serviceType> [-cacheType <cacheType>] [-td <positive_integer>] [-maxClient <positive_integer>] [-maxReq <positive_integer>] [-cacheable (YES | NO)] [-cip (ENABLED | DISABLED) [<cipHeader>]] [-usip (YES | NO)] [-pathMonitor (YES | NO)] [-pathMonitorIndv (YES | NO)] [-useproxyport (YES | NO)] [-healthMonitor (YES | NO)] [-sc (ON | OFF)] [-sp (ON | OFF)] [-rtspSessionidRemap (ON | OFF)] [-cltTimeout <secs>] [-svrTimeout <secs>] [-CKA (YES | NO)] [-TCPB (YES | NO)] [-CMP (YES | NO)] [-maxBandwidth <positive_integer>] [-monThreshold <positive_integer>] [-state (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-autoScale <autoScale> -memberPort <port>]

serviceGroupName

Name of the service group. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the name is created.

serviceType

Protocol used to exchange data with the service.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, RPCSVR, DNS, ADNS, SNMP, RTSP, DHCPRA, ANY, SIP_UDP, DNS_TCP, ADNS_TCP, MYSQL, MSSQL, RADIUS, RDP, DIAMETER, SSL_DIAMETER, TFTP

cacheType

Cache type supported by the cache server.

Possible values: TRANSPARENT, REVERSE, FORWARD

td

Traffic Domain Id.

Maximum value: 4094

maxClient

Maximum number of simultaneous open connections for the service group.

Maximum value: 4294967294

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

Maximum value: 65535

cacheable

Use the transparent cache redirection virtual server to forward the request to the cache server.

Note: Do not set this parameter if you set the Cache Type.

Possible values: YES, NO

Default value: NO

cip

Insert the Client IP header in requests forwarded to the service.

Possible values: ENABLED, DISABLED

cipHeader

Name of the HTTP header whose value must be set to the IP address of the client. Used with the Client IP parameter. If client IP insertion is enabled, and the client IP header is not specified, the value of Client IP Header parameter or the value set by the set ns config command is used as client's IP header name.

usip

Use client's IP address as the source IP address when initiating connection to the server. With the NO setting, which is the default, a mapped IP (MIP) address or subnet IP (SNIP) address is used as the source IP address to initiate server side connections.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions.

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

sc

State of the SureConnect feature for the service group.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service group.

Possible values: ON, OFF

Default value: OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service group.

Possible values: ON, OFF

Default value: OFF

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CKA

Enable client keep-alive for the service group.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service group.

Possible values: YES, NO

CMP

Enable compression for the specified service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

Maximum value: 4294967287

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Maximum value: 65535

state

Initial state of the service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

downStateFlush

Perform delayed clean-up of connections to all services in the service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service group.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service group.

comment

Any information about the service group.

appflowLog

Enable logging of AppFlow information for the specified service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile for the service group.

autoScale

Auto scale option for a servicegroup

Possible values: DISABLED, DNS, POLICY

Default value: NSA_AS_DISABLED

add servicegroup http_svc_group http To add service groups sgrp1, sgrp2 and sgrp3 at one go use the following command: add servicegroup sgrp[1-3] http

---

Removes a service group.

rm serviceGroup <serviceGroupName>@

serviceGroupName

Name of the service group.

rm servicegroup http_svc_group To remove multiple servicegroups at once, the following command can be used: rm servicegroup http_svc_group[1-3]

Modifies the specified parameters of a service group.

set serviceGroup <serviceGroupName>@ [(<serverName>@ <port> [-weight <positive_integer>] [-CustomServerID <string>] [-hashId <positive_integer>]) | -maxClient <positive_integer> | -maxReq <positive_integer> | -cacheable (YES | NO) | -cip (ENABLED | DISABLED) | <cipHeader> | -usip (YES | NO) | -useproxyport (YES | NO) | -sc (ON | OFF) | -sp (ON | OFF) | -

rtspSessionidRemap (ON | OFF) | -cltTimeout <secs> | -svrTimeout <secs> | -CKA (YES | NO) | -TCPB (YES | NO) | -CMP (YES | NO) | -maxBandwidth <positive_integer> | -monThreshold <positive_integer> | -downStateFlush (ENABLED | DISABLED)] [-monitorName <string> -weight <positive_integer>] [-healthMonitor (YES | NO)] [-pathMonitor (YES | NO)] [-pathMonitorIndv (YES | NO)] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>]

serviceGroupName

Name of the service group.

serverName

Name of the server to which to bind the service group.

monitorName

Name of the monitor bound to the service group. Used to assign a weight to the monitor.

maxClient

Maximum number of simultaneous open connections for the service group.

Maximum value: 4294967294

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

Maximum value: 65535

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

cacheable

Use the transparent cache redirection virtual server to forward the request to the cache server.

Note: Do not set this parameter if you set the Cache Type.

Possible values: YES, NO

Default value: NO

cip

Insert the Client IP header in requests forwarded to the service.

Possible values: ENABLED, DISABLED

usip

Use client's IP address as the source IP address when initiating connection to the server. With the NO setting, which is the default, a mapped IP (MIP) address or subnet IP (SNIP) address is used as the source IP address to initiate server side connections.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions.

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

sc

State of the SureConnect feature for the service group.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service group.

Possible values: ON, OFF

Default value: OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service group.

Possible values: ON, OFF

Default value: OFF

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CKA

Enable client keep-alive for the service group.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service group.

Possible values: YES, NO

CMP

Enable compression for the specified service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

Maximum value: 4294967287

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Maximum value: 65535

downStateFlush

Perform delayed clean-up of connections to all services in the service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service group.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service group.

comment

Any information about the service group.

appflowLog

Enable logging of AppFlow information for the specified service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile for the service group.

set servicegroup http_svc_group -maxClient 100 To set the attribute maxclient for multiple servicegroups at once, use the following command: set servicegroup http_svc

---

Removes the attributes of the specified service group. Attributes for which a default value is available revert to their default values..Refer to the set serviceGroup command for meanings of the arguments.

unset serviceGroup <serviceGroupName>@ [<serverName>@ <port> [-weight] [-CustomServerID] [-hashId]] [-maxClient] [-maxReq] [-cacheable] [-cip] [-usip] [-useproxyport] [-sc] [-sp] [-rtspSessionidRemap] [-cltTimeout] [-svrTimeout] [-CKA] [-TCPB] [-CMP] [-maxBandwidth] [-monThreshold] [-tcpProfileName] [-httpProfileName] [-appflowLog] [-netProfile] [-monitorName] [-weight] [-healthMonitor] [-cipHeader] [-pathMonitor] [-pathMonitorIndv] [-downStateFlush] [-comment]

unset servicegroup http_svc_group -maxClient

Binds a service to a service group.

bind serviceGroup <serviceGroupName> ((<IP>@ <port>) | <serverName>@ | ((-monitorName <string>@ [-monState (ENABLED | DISABLED)] [-passive]) | -CustomServerID <string> | -state (ENABLED | DISABLED) | -hashId <positive_integer> | |)) [-weight <positive_integer>]

serviceGroupName

Name of the service group.

IP

IP address of the server that hosts the service. Mutually exclusive with the Server Name parameter.

serverName

Name of the server that hosts the service. Mutually exclusive with the IP address parameter.

port

Port number of the service. Each service must have a unique port number.

monitorName

The name of the service or a service group to which the monitor is to be bound.

weight

CustomServerID

Unique service identifier. Used when the persistency type for the virtual server is set to Custom Server ID.

Default value: "None"

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.

state

Initial state of the service after binding.

Possible values: ENABLED, DISABLED

Default value: ENABLED

hashId

Unique numerical identifier used by hash based load balancing methods to identify a service.

Minimum value: 1

bind servicegroup http_svc_group 10.102.27.153 80 To bind multiple servers to a servicegroup, following command can be used: bind servicegroup http_svc_group 10.10

Unbinds a service or a monitor from a service group.

unbind serviceGroup <serviceGroupName> ((<IP>@ <port>) | <serverName>@ | -monitorName <string>@)

serviceGroupName

Name of the service group.

IP

IP address of the server that hosts the service. Mutually exclusive with the Server Name parameter.

serverName

Name of the server that hosts the service. Mutually exclusive with the IP Address parameter.

port

Port number of the service.

monitorName

Name of the monitor to bind to the service group.

unbind servicegroup http_svc_group 10.102.27.153 80 To unbind multiple servers following command can be used: unbind servicegroup http_svc_group 10.102.27.[153-1

Enables a service group or a member of the service group.

enable serviceGroup <serviceGroupName>@ [<serverName>@ <port>]

serviceGroupName

Name of the service group.

serverName

Name of the server that hosts the service.

port

Port number of the service to be enabled.

enable servicegroup http_svc_group To enable multiple service groups at one go use the following command: enable servicegroup http_svc_group[1-3]

Disables a service group or a member of a service group. To disable a service group, provide only the service group name. To disable only a member of a service group, in addition to the service group name, provide the name of the server that hosts the service, and the port number of the service.

disable serviceGroup <serviceGroupName>@ [<serverName>@ <port>] [-delay <secs>] [-graceFul (YES | NO)]

serviceGroupName

Name of the service group.

serverName

Name of the server that hosts the service.

port

Port number of the service.

delay

Time, in seconds, allocated for a shutdown of the services in the service group. During this period, new requests are sent to the service only for clients who already have persistent sessions on

the appliance. Requests from new clients are load balanced among other available services. After the delay time expires, no requests are sent to the service, and the service is marked as unavailable (OUT OF SERVICE).

graceFul

Wait for all existing connections to the service to terminate before shutting down the service.

Possible values: YES, NO

Default value: NO

disable servicegroup http_svc_group 10.102.27.153 80 -delay 10 To disable multiple servicegroups use the following command: disable servicegroup http_svc_group[1-3

Displays the specified service group's binding information.

show serviceGroup [<serviceGroupName> | -includeMembers]

serviceGroupName

Name of the service group.

includeMembers

Display the members of the listed service groups in addition to their settings. Can be specified when no service group name is provided in the command. In that case, the details displayed for each service group are identical to the details displayed when a service group name is provided, except that bound monitors are not displayed.

summary

fullValues

format

level

numOfconnections

This will tell the number of client side connections are still open.

serviceType

Protocol used to exchange data with the service.

port

The port number of the service to be enabled.

td

Traffic Domain Id.

serviceConfTpye

serviceConfType

The configuration type of the service group.

value

SSL Status.

cacheType

Cache type supported by the cache server.

maxClient

Maximum number of simultaneous open connections for the service group.

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

cacheable

The state of cache on the service.

cip

Insert the Client IP header in requests forwarded to the service.

cipHeader

CIP Header.

usip

Use client's IP address as the source IP address when initiating connection to the server. With the NO setting, which is the default, a mapped IP (MIP) address or subnet IP (SNIP) address is used as the source IP address to initiate server side connections.

pathMonitor

Path monitoring for clustering

pathMonitorIndv

Individual Path monitoring decisions.

useproxyport

The use of client's Port.

monweight

weight of the monitor that is bound to servicegroup.

sc

Whether SureConnect is enabled on this service or not.

sp

Enable surge protection for the service group.

rtspSessionidRemap

Enable RTSP session ID mapping for the service group.

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

CKA

Enable client keep-alive for the service group.

TCPB

Enable TCP buffering for the service group.

CMP

Enable compression for the specified service.

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

state

Monitor state.

svrState

The state of the service

delay

The remaning time in seconds for the service to be disabledNOTE: This attribute is deprecated.Changed from positional to keyword to avoid confusion with serverName

IP

IP Address.

serverName

The name of the server to be changed.

monitorName

Monitor name.

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

monState

The running state of the monitor on this service.

weight

weight of the monitor that is bound to servicegroup.

CustomServerID

The identifier for this IP:Port pair. Used when the persistency type is set to Custom Server ID.

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.NOTE: This attribute is deprecated.Instead of integer now serverId will be a string and you can use -customserverid instead of -serverID.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

monitorTotalProbes

Total number of probes sent to monitor this service.

monitorTotalFailedProbes

Total number of failed probes

monitorCurrentFailedProbes

Total number of currently failed probes

downStateFlush

Perform delayed cleanup of connections on this vserver.

lastresponse

The string form of monstatcode.

stateChangeTimeSec

Time when last state change occurred. Seconds part.

stateChangeTimemSec

Time when last state change occurred. Milliseconds part.

timeSinceLastStateChange

Time in milliseconds since the last state change.NOTE: This attribute is deprecated.This will no longer show the correct information. Use the ticksSinceLastStateChange option instead.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

StateUpdateReason

Checks state update reason on the secondary node.

ClMonOwner

Tells the mon owner of the service.

ClMonView

Tells the view id of the monitoring owner.

groupCount

Servicegroup Count

comment

Any information about the service group.

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service group.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service group.

hashId

The hash identifier for the service. This must be unique for each service. This parameter is used by hash based load balancing methods.

graceFul

Indicates graceful shutdown of the service. System will wait for all outstanding connections to this service to be closed before disabling the service.

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

appflowLog

Enable logging of AppFlow information for the specified service group.

netProfile

Network profile for the service group.

autoScale

Auto scale option for a servicegroup

memberPort

member port

serviceIPstr

This field has been intorduced to show the dbs services ip

serviceGroupEntName2

passive

Indicates if load monitor is passive. A passive load monitor does not remove service from LB decision when threshold is breached.

devno

count

stateflag

Displays configuration statistics of the specified service group or all the service groups configured on the appliance.

stat serviceGroup [<serviceGroupName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

serviceGroupName

Name of the service group for which to display settings.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Service type (Type)

The service type of this service.Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

Renames a service group.

rename serviceGroup <serviceGroupName>@ <newName>@

serviceGroupName

Existing name of the service group.

newName

New name for the service group.

rename service svcgrp1 svcgrp-new1

---

serviceGroupMember

Sep 22, 2015

The following operations can be performed on "serviceGroupMember":

stat serviceGroupMember

Display statistics of a service group member.

Synopsys

stat serviceGroupMember <serviceGroupName> (<IP> | <serverName>) <port> [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

serviceGroupName

Displays statistics for the specified service group.Name of the service group. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my servicegroup" or 'my servicegroup').

IP

IP address of the service group. Mutually exclusive with the server name parameter.

serverName

Name of the server. Mutually exclusive with the IP address parameter.

port

Port number of the service group member.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Average server TTFB (SvrTTFB)

Average TTFB between the NetScaler appliance and the server.TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Service type (Type)

The service type of this service.Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Current client connections (ClntConn)

Number of current client connections.

Requests in surge queue (SurgeQ)

Number of requests in the surge queue.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Connections in reuse pool (ReuseP)

Number of requests in the idle queue/reuse pool.

Maximum server connections (MaxConn)

Maximum open connections allowed on this service.

servicegroupbindings

Sep 22, 2015

The following operations can be performed on "servicegroupbindings":

show servicegroupbindings

Displays servicegroup information followed by vservers bound to it.

Synopsys

show servicegroupbindings <serviceGroupName>

Arguments

serviceGroupName

The name of the service.

Outputs

IPAddress

The IP address of the vserver.

port

The port of the vserver.

state

The state of the service group

svrState

The state of the vserver

vServerName

The name of the vserver.

stateflag

devno

count

svcbindings

Sep 22, 2015

The following operations can be performed on "svcbindings":

show svcbindings

Displays a list of all virtual servers to which the service is bound.

Synopsys

show svcbindings <serviceName>

Arguments

serviceName

The name of the service.

Outputs

IPAddress

The IP address of the vserver.

port

The port of the vserver.

svrState

The state of the vserver

vServerName

The name of the vserver.

stateflag

devno

count

uiinternal

Sep 22, 2015

The following operations can be performed on "uiinternal":

set | unset | show

set uiinternal

set uiinternal data for the entities

Synopsys

set uiinternal <entityType> <name> [-template <string>] [-comment <string>] [-rule <string>]

Arguments

entityType

The entitiy type of UI internal data

Possible values: LBVSERVER, GSLBVSERVER, CRVSERVER, VPNVSERVER, CSVSERVER, AUTHENTICATIONVSERVER, SERVER, SERVICE, SERVICEGROUP, GSLBSERVICE, EXPRESSION, VPNURL

name

The entity name

template

The application template associated with entity

comment

The application template associated with entity

rule

rules associated with entity

Example

```
set uiinternal lbvserver v1 -template app1
```

---

unset uiinternal

unset uiinternal for the entities.Refer to the set uiinternal command for meanings of the arguments.

Synopsys

unset uiinternal <entityType> <name> [-template] [-comment] [-rule] [-all]

Example

unset uiinternal lbvserver v1 -template app1

---

show uiinternal

display all UI internal data information for the entities

Synopsys

show uiinternal [<entityType>] [<name>]

Arguments

entityType

The entitiy type of UI internal data

Possible values: LBVSERVER, GSLBVSERVER, CRVSERVER, VPNVSERVER, CSVSERVER, AUTHENTICATIONVSERVER, SERVER, SERVICE, SERVICEGROUP, GSLBSERVICE, EXPRESSION, VPNURL

name

The entity name

summary

fullValues

format

level

Outputs

template

The template associated with the entity

comment

The comment associated with the entity

uiinfo

The uiinfo associated with the entity

rule

The rule associated with the entity

devno

count

stateflag

Example

show uiinternal LBVSERVER v1

vserver

Sep 22, 2015

The following operations can be performed on "vserver":

rm | set | unset | enable | disable | show

rm vserver

Use this command to remove a virtual server. NOTE: This command is deprecated.This command is deprecated in 10.0, instead you can use commands such as rm lb vserver

Synopsys

Arguments

name

The name of the virtual server to be removed.

Example

```
rm vserver lb_vip  To remove multiple vservers, use the following command:  rm vserver lb_vip[1-3]
```

---

set vserver

Use this command to modify the parameters for an existing virtual server. NOTE: This command is deprecated.This command is deprecated in 10.0, instead you can use commands such as set lb vserver

Synopsys

Arguments

name

The name of the virtual server for which the parameters are to be set.

backupVServer

The name of the backup virtual server for this virtual server.

redirectURL

The URL where traffic is redirected if the virtual server in the system becomes unavailable.

cacheable

Use this option to specify whether a virtual server (used for load balancing or content switching) routes requests to the cache redirection virtual server before sending it to the configured servers.

Possible values: YES, NO

cltTimeout

The timeout value in seconds for idle client connection

Maximum value: 31536000

soMethod

The spillover factor. The system will use this value to determine if it should send traffic to the backupvserver when the main virtual server reaches the spillover threshold.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

The state of the spillover persistence.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

The spillover persistence entry timeout.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

The spillver threshold value.

Minimum value: 1

Maximum value: 4294967294

pushVserver

The lb vserver of type PUSH/SSL_PUSH to which server pushes the updates received on the client facing non-push lb vserver.

Example

set vserver lb_vip -backupVServerName bkvip_lbvip To set backup vserver for multiple vservers at once, use the following command: set vserver lb_vip[1-3] -backupVServ

---

unset vserver

Use this command to unset the backup virtual server or the redirectURL that has been set on the virtual server..Refer to the set vserver command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

Example

unset vserver lb_vip -backupVServer To unset the backup vserver for multiple vservers use the following command: unset vserver lb_vip[1-3] -backupVServer

---

enable vserver

Use this command to enable a virtual server. Note: Virtual servers, when added, are enabled by default. NOTE: This command is deprecated.This command is deprecated in 10.0, instead you can use commands such as enable lb vserver

Synopsys

Arguments

name

The name of the virtual server to be enabled.

Example

enable vserver lb_vip To enable multiple vservers, use the following command: enable vserver lb_vip[1-3]

---

disable vserver

Use this command to disable (take out of service) a virtual server. NOTE: This command is deprecated.This command is deprecated in 10.0, instead you can use commands such as disable lb vserver

Synopsys

Arguments

name

The name of the virtual server to be disabled.

Notes:

1. The system will continue to respond to ARP and/or ping requests for the IP address of this virtual server.

2. As the virtual server is still configured in the system, you can enable the virtual server using the ###enable vserver### command.

Example

disable vserver lb_vip To disable multiple vservers, use the following command: disable vserver lb_vip[1-3]

---

show vserver

Displays information about all virtual servers configured on the appliance.

Synopsys

show vserver

Example

show vserver lb_vip

Cache Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- cache
- cache contentGroup
- cache forwardProxy
- cache global
- cache object
- cache parameter
- cache policy
- cache policylabel
- cache selector
- cache stats

cache

Sep 22, 2015

The following operations can be performed on "cache":

stat cache

Shows Integrated Cache performance statistics.

Synopsys

stat cache [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

Maximum memory(KB) Deprecated (MaxMem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Recent successful reval ratio(%) (RPSucRev)

Recently recorded percentage of times stored content was successfully revalidated by a 304 response rather than by a full response

Recent storable miss ratio(%) (RPctStMis)

Recently recorded ratio of store-able misses to all misses expressed as percentage.

Recent parameterized 304 hit ratio(%) (RPPHit)

Recently recorded ratio of parameterized 304 hits to all parameterized hits expressed as a percentage

Recent origin bandwidth saved(%) (RPOrBan)

Bytes served from cache divided by total bytes served to client. This ratio can be greater than 1 because of the assumption that all compression has been done in the NetScaler.

Recent hit ratio(%) (RPctHit)

Recently recorded cache hit ratio expressed as percentage

Recent byte hit ratio(%) (RPcByHit)

Recently recorded cache byte hit ratio expressed as percentage. Here we define byte hit ratio as ((number of bytes served from the cache)/(total number of bytes served to the client)). This is the standard definition of Byte Hit

Ratio. If compression is turned ON in NS then this ratio doesn't mean much. This might under or over estimate the origin-to-cache bandwidth saving (depending upon whether bytes served by CMP in NetScaler are more or less than compressed bytes served from the cache). If CMP is turned OFF in NS then this ratio is same as cacheRecentPercentOriginBandwidthSaved.

Recent 304 hit ratio(%) (RPct304Hit)

Recently recorded ratio of 304 hits to all hits expressed as percentage

Utilized memory(KB) (UtiMem)

Amount of memory the integrated cache is currently using.

Maximum memory active value(KB) (MaxMemActive)

Currently active value of maximum memory

Maximum memory(KB) (Max64Mem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Poll every time hit ratio(%) (PPeHit)

Percentage of cache hits in content groups that have Poll Every Time enabled, relative to all searches of content groups with Poll Every Time enabled.

Poll every time hits (PeHit)

Number of times a cache hit was found during a search of a content group that has Poll Every Time enabled.

Parameterized 304 hit ratio(%) (PP304Hit)

Percentage of parameterized 304 hits relative to all parameterized hits.

Total parameterized hits (PHit)

Parameterized requests resulting in either a 304 or non-304 hit.

Successful reval ratio(%) (PSucRev)

Percentage of times stored content was successfully revalidated by a 304 (Object Not Modifed) response rather than by a full response

Storable miss ratio(%) (PStrMiss)

Responses that were fetched from the origin, stored in the cache, and then served to the client, as a percentage of all cache misses.

Conversions to conditional req (FuToCon)

Number of user-agent requests for a cached Poll Every Time (PET) response that were sent to the origin server as conditional requests.

Successful revalidations (TSucRev)

Total number of times stored content was successfully revalidated by a 304 Not Modified response from the origin.

Revalidations (Reval)

Responses that an intervening cache revalidated with the integrated cache before serving, as determined by a Cache-Control: Max-Age header configurable in the integrated cache

Non-storable misses (NStrMiss)

Cache misses for which the fetched response is not stored in the cache. These responses match policies with a NOCACHE action or are affected by Poll Every Time.

Storable misses (StrMiss)

Cache misses for which the fetched response is stored in the cache before serving it to the client. Storable misses conform to a built-in or user-defined caching policy that contains a CACHE action.

Compressed bytes from cache (CmpBySer)

Number of compressed bytes served from the cache

Byte hit ratio(%) (PByHit)

Bytes served from the cache divided by total bytes served to the client. If compression is On in the NetScaler, this ratio may not reflect the bytes served by the compression module. If the compression is Off, this ratio is the same as cachePercentOriginBandwidthSaved.

Bytes served by cache (BySer)

Total number of bytes served from the integrated cache

Bytes served by NetScaler (RespBy)

Total number of HTTP response bytes served by NetScaler from both the origin and the cache

304 hit ratio(%) (Pct304Hit)

304 responses as a percentage of all responses that the NetScaler served.

Marker objects (NumMark)

Marker objects created when a response exceeds the maximum or minimum size for entries in its content group or has not yet received the minimum number of hits required for items in its content group.

Origin bandwidth saved(%) (POrBan)

Percentage of origin bandwidth saved, expressed as number of bytes served from the integrated cache divided by all bytes served. The assumption is that all compression is done in the NetScaler.

Hit ratio(%) (PctHit)

Cache hits as percentage of the total number of requests

Misses (TotMiss)

Intercepted HTTP requests requiring fetches from origin server.

Hits (TotHit)

Responses served from the integrated cache. These responses match a policy with a CACHE action.

Requests (CacReq)

Total cache hits plus total cache misses.

Cached objects (NumCac)

Responses currently in integrated cache. Includes responses fully downloaded, in the process of being downloaded, and expired or flushed but not yet removed.

Objects saved on disk (NumObjCacDsk)

Cached responses currently saved on disk. Includes responses fully saved to disk, and expired or flushed but not yet removed.

MB read from disk (NumMBReadsDisk)

Total Number of MB read from disk since last reboot.

MB written to disk (NumMBWritesDisk)

Total Number of MB written to disk since last reboot.

Hits being served (CacHit)

This number should be close to the number of hits being served currently.

Misses being handled (CurMiss)

Responses fetched from the origin and served from the cache. Should approximate storable misses. Does not include non-storable misses.

Non-304 hits (Non304Hit)

Total number of full (non-304) responses served from the cache. A 304 status code indicates that a response has not been modified since the last time it was served

304 hits (304Hit)

Object not modified responses served from the cache. (Status code 304 served instead of the full response.)

sql hits (sqlHit)

sql response served from cache

Expire at last byte (ExpLa)

Instances of content expiring immediately after receiving the last body byte due to the Expire at Last Byte setting for the content group.

Flashcache misses (FlMi)

Number of requests to a content group with flash cache enabled that were cache misses. Flash cache distributes the response to all the clients in aqueue.

Flashcache hits (FlHi)

Number of requests to a content group with flash cache enabled that were cache hits. The flash cache setting queues requests that arrive simultaneously and distributes the response to all the clients in the queue.

Parameterized inval requests (PInReq)

Requests matching a policy with an invalidation (INVAL) action and a content group that uses an invalidation selector or parameters.

Full inval requests (NPInReq)

Requests that match an invalidation policy where the invalGroups parameter is configured and expires one or more content groups.

Inval requests (INStrMis)

Requests that match an invalidation policy and result in expiration of specific cached responses or entire content groups.

Parameterized requests (PReq)

Total number of requests where the content group has hit and invalidation parameters or selectors.

Parameterized non-304 hits (PN304Hit)

Parameterized requests resulting in a full response (not status code 304: Object Not Updated) served from the cache.

Parameterized 304 hits (P304Hit)

Parameterized requests resulting in an object not modified (status code 304) response.

Poll every time requests (PeReq)

Requests that triggered a search of a content group that has Poll Every Time (PET) enabled (always consult the origin server before serving cached data).

Memory allocation failures (ErrMem)

Total number of times the cache failed to allocate memory to store responses.

Largest response so far(B) (LarResp)

Size, in bytes, of largest response sent to client from the cache or the origin server.

MB saved on disk (NumMBCacDsk)

Size (MB) of cached responses currently saved on disk. Includes responses fully saved to disk, and expired or flushed but not yet removed.

Compressed bytes transmitted

Number of bytes the NetScaler sends to the client after compressing the response from the server.

Compressible bytes received

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

cache contentGroup

Sep 22, 2015

The following operations can be performed on "cache contentGroup":

add | rm | set | unset | show | expire | flush

add cache contentGroup

Creates a new content group for grouping cached objects on the basis of some unique property.

Synopsys

add cache contentGroup <name> [-weakPosRelExpiry <secs> | -relExpiry <secs> | -relExpiryMilliSec <msecs> | -absExpiry <HH:MM> ... | -absExpiryGMT <HH:MM> ...] [-heurExpiryParam <positive_integer>] [-weakNegRelExpiry <secs>] [(-hitParams <string> ... [-ignoreParamValueCase (YES | NO) | -hitSelector <string> | -invalSelector <string>] [-matchCookies (YES | NO)])] [-invalParams <string> ... [-invalRestrictedToHost (YES | NO)]] [-pollEveryTime (YES | NO)] [-ignoreReloadReq (YES | NO)] [-removeCookies (YES | NO)] [-prefetch (YES | NO) [-prefetchPeriod <secs> | -prefetchPeriodMilliSec <msecs>]] [-prefetchMaxPending <positive_integer>] [-flashCache (YES | NO)] [-expireAtLastByte (YES | NO)] [-insertVia (YES | NO)] [-insertAge (YES | NO)] [-insertETag (YES | NO)] [-cacheControl <string>] [-quickAbortSize <KBytes>] [-minResSize <KBytes>] [-maxResSize <KBytes>] [-memLimit <MBytes>] [-ignoreReqCachingHdrs (YES | NO)] [-minHits <integer>] [-alwaysEvalPolicies (YES | NO)] [-pinned (YES | NO)] [-lazyDnsResolve (YES | NO)] [-type <type>]

Arguments

name

Name for the content group. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the content group is created.

weakPosRelExpiry

Relative expiry time, in seconds, for expiring positive responses with response codes between 200 and 399. Cannot be used in combination with other Expiry attributes. Similar to -relExpiry but has lower precedence.

Default value: VAL_NOT_SET

Maximum value: 31536000

heurExpiryParam

Heuristic expiry time, in percent of the duration, since the object was last modified.

Default value: VAL_NOT_SET

Maximum value: 100

relExpiry

Relative expiry time, in seconds, after which to expire an object cached in this content group.

Default value: VAL_NOT_SET

Maximum value: 31536000

relExpiryMilliSec

Relative expiry time, in milliseconds, after which to expire an object cached in this content group.

Default value: VAL_NOT_SET

Maximum value: 86400000

absExpiry

Local time, up to 4 times a day, at which all objects in the content group must expire.

CLI Users:

For example, to specify that the objects in the content group should expire by 11:00 PM, type the following command: add cache contentgroup <contentgroup name> -absexpiry 23:00

To specify that the objects in the content group should expire at 10:00 AM, 3 PM, 6 PM, and 11:00 PM, type: add cache contentgroup <contentgroup name> -absexpiry 10:00 15:00 18:00 23:00

absExpiryGMT

Coordinated Universal Time (GMT), up to 4 times a day, when all objects in the content group must expire.

weakNegRelExpiry

Relative expiry time, in seconds, for expiring negative responses. This value is used only if the expiry time cannot be determined from any other source. It is applicable only to the following status codes: 307, 403, 404, and 410.

Default value: VAL_NOT_SET

Maximum value: 31536000

hitParams

Parameters to use for parameterized hit evaluation of an object. Up to 128 parameters can be specified. Mutually exclusive with the Hit Selector parameter.

invalParams

Parameters for parameterized invalidation of an object. You can specify up to 8 parameters. Mutually exclusive with invalSelector.

ignoreParamValueCase

Ignore case when comparing parameter values during parameterized hit evaluation. (Parameter value case is ignored by default during parameterized invalidation.)

Possible values: YES, NO

Default value: VAL_NOT_SET

matchCookies

Evaluate for parameters in the cookie header also.

Possible values: YES, NO

Default value: VAL_NOT_SET

invalRestrictedToHost

Take the host header into account during parameterized invalidation.

Possible values: YES, NO

Default value: VAL_NOT_SET

pollEveryTime

Always poll for the objects in this content group. That is, retrieve the objects from the origin server whenever they are requested.

Possible values: YES, NO

Default value: NO

ignoreReloadReq

Ignore any request to reload a cached object from the origin server.

To guard against Denial of Service attacks, set this parameter to YES. For RFC-compliant behavior, set it to NO.

Possible values: YES, NO

Default value: YES

removeCookies

Remove cookies from responses.

Possible values: YES, NO

Default value: YES

prefetch

Attempt to refresh objects that are about to go stale.

Possible values: YES, NO

Default value: YES

prefetchPeriod

Time period, in seconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: VAL_NOT_SET

Maximum value: 4294967294

prefetchPeriodMilliSec

Time period, in milliseconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: VAL_NOT_SET

Maximum value: 4294967290

prefetchMaxPending

Maximum number of outstanding prefetches that can be queued for the content group.

Default value: VAL_NOT_SET

Maximum value: 4294967294

flashCache

Perform flash cache. Mutually exclusive with Poll Every Time (PET) on the same content group.

Possible values: YES, NO

Default value: NO

expireAtLastByte

Force expiration of the content immediately after the response is downloaded (upon receipt of the last byte of the response body). Applicable only to positive responses.

Possible values: YES, NO

Default value: NO

insertVia

Insert a Via header into the response.

Possible values: YES, NO

Default value: YES

insertAge

Insert an Age header into the response. An Age header contains information about the age of the object, in seconds, as calculated by the integrated cache.

Possible values: YES, NO

Default value: YES

insertETag

Insert an ETag header in the response. With ETag header insertion, the integrated cache does not serve full responses on repeat requests.

Possible values: YES, NO

Default value: YES

cacheControl

Insert a Cache-Control header into the response.

quickAbortSize

If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.

Default value: 4194303

Maximum value: 4194303

minResSize

Minimum size of a response that can be cached in this content group.

Default minimum response size is 0.

Maximum value: 2097151

maxResSize

Maximum size of a response that can be cached in this content group.

Default value: 80

Maximum value: 2097151

memLimit

Maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

Default value: 65536

ignoreReqCachingHdrs

Ignore Cache-Control and Pragma headers in the incoming request.

Possible values: YES, NO

Default value: YES

minHits

Number of hits that qualifies a response for storage in this content group.

alwaysEvalPolicies

Force policy evaluation for each response arriving from the origin server. Cannot be set to YES if the Prefetch

parameter is also set to YES.

Possible values: YES, NO

Default value: NO

pinned

Do not flush objects from this content group under memory pressure.

Possible values: YES, NO

Default value: NO

lazyDnsResolve

Perform DNS resolution for responses only if the destination IP address in the request does not match the destination IP address of the cached response.

Possible values: YES, NO

Default value: YES

hitSelector

Selector for evaluating whether an object gets stored in a particular content group. A selector is an abstraction for a collection of PIXL expressions.

invalSelector

Selector for invalidating objects in the content group. A selector is an abstraction for a collection of PIXL expressions.

type

The type of the content group.

Possible values: HTTP, MYSQL, MSSQL

Default value: NSSVC_HTTP

---

rm cache contentGroup

Removes the specified content group. Before removing, make sure that no cache policy has its storeInGroup attribute set to this group, otherwise the group cannot be removed.

Synopsys

rm cache contentGroup <name>

Arguments

name

Name of the content group to be removed.

---

set cache contentGroup

Modifies the specified attributes of the content group.

Synopsys

set cache contentGroup <name> [-weakPosRelExpiry <secs> | -relExpiry <secs> | -relExpiryMilliSec <msecs> | -absExpiry <HH:MM> ... | -absExpiryGMT <HH:MM> ...] [-heurExpiryParam <positive_integer>] [-weakNegRelExpiry <secs>] [-hitParams <string> ... | -hitSelector <string> | -invalSelector <string>] [-invalParams <string> ...] [-ignoreParamValueCase (YES | NO)] [-matchCookies (YES | NO)] [-invalRestrictedToHost (YES | NO)] [-pollEveryTime (YES | NO)] [-ignoreReloadReq (YES | NO)] [-removeCookies (YES | NO)] [-prefetch (YES | NO)] [-prefetchPeriod <secs> | -prefetchPeriodMilliSec <msecs>] [-prefetchMaxPending <positive_integer>] [-flashCache (YES | NO)] [-expireAtLastByte (YES | NO)] [-insertVia (YES | NO)] [-insertAge (YES | NO)] [-insertETag (YES | NO)] [-cacheControl <string>] [-quickAbortSize <KBytes>] [-minResSize <KBytes>] [-maxResSize <KBytes>] [-memLimit <MBytes>] [-ignoreReqCachingHdrs (YES | NO)] [-minHits <integer>] [-alwaysEvalPolicies (YES | NO)] [-pinned (YES | NO)] [-lazyDnsResolve (YES | NO)]

Arguments

name

Name of the content group to be modified.

weakPosRelExpiry

Relative expiry time, in seconds, for expiring positive responses with response codes between 200 and 399. Cannot be used in combination with other Expiry attributes. Similar to -relExpiry but has lower precedence.

Maximum value: 31536000

heurExpiryParam

Heuristic expiry time, in percent of the duration, since the object was last modified.

Maximum value: 100

relExpiry

Relative expiry time, in seconds, after which to expire an object cached in this content group.

Default value: VAL_NOT_SET

Maximum value: 31536000

relExpiryMilliSec

Relative expiry time, in milliseconds, after which to expire an object cached in this content group.

Default value: VAL_NOT_SET

Maximum value: 86400000

absExpiry

Local time, up to 4 times a day, at which all objects in the content group must expire.

CLI Users:

For example, to specify that the objects in the content group should expire by 11:00 PM, type the following command: add cache contentgroup <contentgroup name> -absexpiry 23:00

To specify that the objects in the content group should expire at 10:00 AM, 3 PM, 6 PM, and 11:00 PM, type: add cache contentgroup <contentgroup name> -absexpiry 10:00 15:00 18:00 23:00

absExpiryGMT

Coordinated Universal Time (GMT), up to 4 times a day, when all objects in the content group must expire.

weakNegRelExpiry

Relative expiry time, in seconds, for expiring negative responses. This value is used only if the expiry time cannot be determined from any other source. It is applicable only to the following status codes: 307, 403, 404, and 410.

Maximum value: 31536000

hitParams

Parameters to use for parameterized hit evaluation of an object. Up to 128 parameters can be specified. Mutually exclusive with the Hit Selector parameter.

invalParams

Parameters for parameterized invalidation of an object. You can specify up to 8 parameters. Mutually exclusive with invalSelector.

ignoreParamValueCase

Ignore case when comparing parameter values during parameterized hit evaluation. (Parameter value case is ignored by default during parameterized invalidation.)

Possible values: YES, NO

matchCookies

Evaluate for parameters in the cookie header also.

Possible values: YES, NO

invalRestrictedToHost

Take the host header into account during parameterized invalidation.

Possible values: YES, NO

pollEveryTime

Always poll for the objects in this content group. That is, retrieve the objects from the origin server whenever they are requested.

Possible values: YES, NO

Default value: NO

ignoreReloadReq

Ignore any request to reload a cached object from the origin server.

To guard against Denial of Service attacks, set this parameter to YES. For RFC-compliant behavior, set it to NO.

Possible values: YES, NO

Default value: YES

removeCookies

Remove cookies from responses.

Possible values: YES, NO

Default value: YES

prefetch

Attempt to refresh objects that are about to go stale.

Possible values: YES, NO

Default value: YES

prefetchPeriod

Time period, in seconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: VAL_NOT_SET

Maximum value: 4294967294

prefetchPeriodMilliSec

Time period, in milliseconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: VAL_NOT_SET

Maximum value: 4294967290

prefetchMaxPending

Maximum number of outstanding prefetches that can be queued for the content group.

Maximum value: 4294967294

flashCache

Perform flash cache. Mutually exclusive with Poll Every Time (PET) on the same content group.

Possible values: YES, NO

Default value: NO

expireAtLastByte

Force expiration of the content immediately after the response is downloaded (upon receipt of the last byte of the response body). Applicable only to positive responses.

Possible values: YES, NO

Default value: NO

insertVia

Insert a Via header into the response.

Possible values: YES, NO

Default value: YES

insertAge

Insert an Age header into the response. An Age header contains information about the age of the object, in seconds, as calculated by the integrated cache.

Possible values: YES, NO

Default value: YES

insertETag

Insert an ETag header in the response. With ETag header insertion, the integrated cache does not serve full responses on repeat requests.

Possible values: YES, NO

Default value: YES

cacheControl

Insert a Cache-Control header into the response.

quickAbortSize

If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.

Maximum value: 4194303

minResSize

Minimum size of a response that can be cached in this content group.

Default minimum response size is 0.

Maximum value: 2097151

maxResSize

Maximum size of a response that can be cached in this content group.

Default value: 80

Maximum value: 2097151

memLimit

Maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

Default value: 65536

ignoreReqCachingHdrs

Ignore Cache-Control and Pragma headers in the incoming request.

Possible values: YES, NO

Default value: YES

minHits

Number of hits that qualifies a response for storage in this content group.

alwaysEvalPolicies

Force policy evaluation for each response arriving from the origin server. Cannot be set to YES if the Prefetch parameter is also set to YES.

Possible values: YES, NO

Default value: NO

pinned

The option for IC from flushing objects from this contentgroup under memory pressure. Set YES for IC to take this state.

Possible values: YES, NO

Default value: NO

lazyDnsResolve

Perform DNS resolution for responses only if the destination IP address in the request does not match the destination IP address of the cached response.

Possible values: YES, NO

Default value: YES

hitSelector

Selector for evaluating whether an object gets stored in a particular content group. A selector is an abstraction for a collection of PIXL expressions.

invalSelector

Selector for invalidating objects in the content group. A selector is an abstraction for a collection of PIXL expressions.

---

unset cache contentGroup

Use this command to remove cache contentGroup settings.Refer to the set cache contentGroup command for meanings of the arguments.

Synopsys

unset cache contentGroup <name> [-weakPosRelExpiry] [-heurExpiryParam] [-relExpiry] [-relExpiryMilliSec] [-absExpiry] [-absExpiryGMT] [-weakNegRelExpiry] [-hitParams] [-invalParams] [-ignoreParamValueCase] [-matchCookies] [-invalRestrictedToHost] [-pollEveryTime] [-ignoreReloadReq] [-removeCookies] [-prefetch] [-prefetchPeriod] [-prefetchPeriodMilliSec] [-prefetchMaxPending] [-flashCache] [-expireAtLastByte] [-insertVia] [-insertAge] [-insertETag] [-cacheControl] [-quickAbortSize] [-minResSize] [-maxResSize] [-memLimit] [-ignoreReqCachingHdrs] [-minHits] [-alwaysEvalPolicies] [-pinned] [-lazyDnsResolve] [-hitSelector] [-invalSelector]

---

show cache contentGroup

Displays information about all content groups, or about the specified content group.

Synopsys

show cache contentGroup [<name>]

Arguments

name

Name of the content group about which to display information.

summary

fullValues

format

level

Outputs

flags

Flags.

type

The type of the content group.

relExpiry

The relative expiry time in seconds.

relExpiryMilliSec

Relative expiry time, in milliseconds, after which to expire an object cached in this content group.

absExpiry

Local time, up to 4 times a day, at which all objects in the content group must expire.

CLI Users:

For example, to specify that the objects in the content group should expire by 11:00 PM, type the following command: add cache contentgroup <contentgroup name> -absexpiry 23:00

To specify that the objects in the content group should expire at 10:00 AM, 3 PM, 6 PM, and 11:00 PM, type: add cache contentgroup <contentgroup name> -absexpiry 10:00 15:00 18:00 23:00

absExpiryGMT

Coordinated Universal Time (GMT), up to 4 times a day, when all objects in the content group must expire.

heurExpiryParam

Heuristic expiry time, in percent of the duration, since the object was last modified.

weakPosRelExpiry

Relative expiry time, in seconds, for expiring positive responses with response codes between 200 and 399. Cannot be used in combination with other Expiry attributes. Similar to -relExpiry but has lower precedence.

weakNegRelExpiry

Relative expiry time, in seconds, for expiring negative responses. This value is used only if the expiry time cannot be determined from any other source. It is applicable only to the following status codes: 307, 403, 404, and 410.

hitParams

Parameters to use for parameterized hit evaluation of an object. Up to 128 parameters can be specified. Mutually exclusive with the Hit Selector parameter.

invalParams

Parameters for parameterized invalidation of an object. You can specify up to 8 parameters. Mutually exclusive with invalSelector.

ignoreParamValueCase

Ignore case when comparing parameter values during parameterized hit evaluation. (Parameter value case is ignored by default during parameterized invalidation.)

matchCookies

Evaluate for parameters in the cookie header also.

invalRestrictedToHost

Take the host header into account during parameterized invalidation.

pollEveryTime

Always poll for the objects in this content group. That is, retrieve the objects from the origin server whenever they are requested.

ignoreReloadReq

Ignore any request to reload a cached object from the origin server.

To guard against Denial of Service attacks, set this parameter to YES. For RFC-compliant behavior, set it to NO.

removeCookies

Remove cookies from responses.

prefetch

Attempt to refresh objects that are about to go stale.

prefetchPeriod

Time period, in seconds before an object's calculated expiry time, during which to attempt prefetch.

prefetchPeriodMilliSec

Time period, in milliseconds before an object's calculated expiry time, during which to attempt prefetch.

prefetchCur

Current outstanding prefetches.

prefetchMaxPending

Maximum number of outstanding prefetches that can be queued for the content group.

flashCache

Perform flash cache. Mutually exclusive with Poll Every Time (PET) on the same content group.

expireAtLastByte

Force expiration of the content immediately after the response is downloaded (upon receipt of the last byte of the response body). Applicable only to positive responses.

insertVia

Insert a Via header into the response.

insertAge

Insert an Age header into the response. An Age header contains information about the age of the object, in seconds, as calculated by the integrated cache.

insertETag

Insert an ETag header in the response. With ETag header insertion, the integrated cache does not serve full responses on repeat requests.

cacheControl

Insert a Cache-Control header into the response.

quickAbortSize

If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.

minResSize

Minimum size of a response that can be cached in this content group.

Default minimum response size is 0.

maxResSize

Maximum size of a response that can be cached in this content group.

memUsage

Current memory usage.

memLimit

Maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

ignoreReqCachingHdrs

Ignore Cache-Control and Pragma headers in the incoming request.

cacheNon304Hits

Cache non 304 hits.

cache304Hits

Cache 304 hits.

cacheCells

Number of cells.

cacheGroupIncarnation

Cache group incarnation.

minHits

Number of hits that qualifies a response for storage in this content group.

alwaysEvalPolicies

Force policy evaluation for each response arriving from the origin server. Cannot be set to YES if the Prefetch parameter is also set to YES.

persist

Setting persist to YES causes IC to save objects in contentgroup to disk.

pinned

Do not flush objects from this content group under memory pressure.

lazyDnsResolve

Perform DNS resolution for responses only if the destination IP address in the request does not match the destination IP address of the cached response.

hitSelector

Selector for evaluating whether an object gets stored in a particular content group. A selector is an abstraction for a collection of PIXL expressions.

invalSelector

Selector for invalidating objects in the content group. A selector is an abstraction for a

collection of PIXL expressions.

policyName

Active cache policies refering to this group.

cacheNumInvalPolicy

Number of active Invalidation policies refering to this group.

markerCells

Numbers of marker cells in this group.

builtin

devno

count

stateflag

---

expire cache contentGroup

Forces expiration of all the objects in the specified content group. The next request for any object in the group is sent to the origin server.

Synopsys

expire cache contentGroup <name>

Arguments

name

Name of the content group whose objects are to be expired.

---

flush cache contentGroup

Flush the objects in the specified content group.

Synopsys

flush cache contentGroup <name> [-query <string> | -selectorValue <string>] [-host <string>]

Arguments

name

Name of the content group from which to flush objects, or "all"

to flush all content groups.

query

Query string specifying individual objects to flush from this group by using parameterized invalidation. If this parameter is not set, all objects are flushed from the group.

host

Flush only objects that belong to the specified host. Do not use except with parameterized invalidation. Also, the Invalidation Restricted to Host parameter for the group must be set to YES.

selectorValue

Value of the selector to be used for flushing objects from the content group. Requires that an invalidation selector be configured for the content group.

cache forwardProxy

Sep 22, 2015

The following operations can be performed on "cache forwardProxy":

add | rm | show

add cache forwardProxy

Allows the cache to act as a forward proxy for other NetScaler appliances or cache servers.

Synopsys

add cache forwardProxy <IPAddress> <port>

Arguments

IPAddress

IP address of the NetScaler appliance or a cache server for which the cache acts as a proxy. Requests coming to the NetScaler with the configured IP address are forwarded to the particular address, without involving the Integrated Cache in any way.

port

Port on the NetScaler appliance or a server for which the cache acts as a proxy

Minimum value: 1

---

rm cache forwardProxy

Removes the forward proxy address from the Integrated Cache. The cache does not act as a proxy to the specified IP address.

Synopsys

rm cache forwardProxy <IPAddress> <port>

Arguments

IPAddress

IP address of the NetScaler appliance or a server for which the cache was as a proxy.

port

Port on the NetScaler appliance or a server for which the cache acts as a proxy

Minimum value: 1

---

show cache forwardProxy

Displays the IP address and the corresponding ports for which the cache acted as a forward proxy.

Synopsys

show cache forwardProxy

Arguments

summary

fullValues

format

level

Outputs

IPAddress

IP address of the NetScaler appliance or a cache server for which the cache acts as a proxy. Requests coming to the NetScaler with the configured IP address are forwarded to the particular address, without involving the Integrated Cache in any way.

port

Forward proxy port.

devno

count

stateflag

cache global

Sep 22, 2015

The following operations can be performed on "cache global":

bind | unbind | show

bind cache global

Binds the cache policy to one of the two global bind points (an unnamed policy label invoked at request time and an unnamed policy label invoked at the response time). The flow type of the policy implicitly determines which label it gets bound to. A policy becomes active only when it is bound. A globally bound policy, it is available to all virtual servers on the NetScaler appliance. All HTTP traffic is evaluated against the global policy labels. Each label contains an ordered list ordered by policies' priority values.

Synopsys

bind cache global <policy> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-type <type>] [-invoke (<labelType> <labelName>)]

Arguments

policy

Name of the policy to bind. (A policy must be created before it can be bound.)

---

unbind cache global

Deactivate the policy by unbinding it from a global bind point.

Synopsys

unbind cache global <policy> [-type <type>] [-priority <positive_integer>]

Arguments

policy

Name of the policy to unbind.

priority

Priority of the NOPOLICY to be unbound. Required only you want to unbind a NOPOLICY that might have been bound to this policy label.

Minimum value: 1

Maximum value: 2147483647

---

show cache global

Displays the global bindings for cache policies.

Synopsys

show cache global [-type <type>]

Arguments

type

The bind point to which policy is bound. When you specify the type, detailed information about that bind point appears.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

summary

fullValues

format

level

Outputs

policyName

Name of the cache policy.NOTE: This attribute is deprecated.Replaced by Policy field

policy

Name of the cache policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority. Applicable only to default-syntax policies.

labelType

Type of policy label to invoke.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE. (To invoke a label associated with a virtual server, specify the name of the virtual server.)

numpol

The number of policies bound to the bindpoint.

flowType

flowtype of the bound cache policy.

rule

The request/response rule that will trigger the given action.NOTE: This attribute is deprecated.

action

The integrated cache action to be applied when the system sees content that matches the rules.NOTE: This attribute is deprecated.

storeInGroup

The content group to store the object when the action directive is CACHE.NOTE: This attribute is deprecated.

invalGroups

The content group(s) to be invalidated when the action directive is INVAL.NOTE: This attribute is deprecated.

invalObjects

The content group(s) whose objects will be invalidated when the action directive is INVAL.

hits

Hits.NOTE: This attribute is deprecated.

flags

Flags.NOTE: This attribute is deprecated.

precedeDefRules

Override the default request/response cacheability rules.NOTE: This attribute is deprecated.

stateflag

devno

count

Example

cache object

Sep 22, 2015

The following operations can be performed on "cache object":

show | expire | flush

show cache object

Displays a list of all cached objects. The list displays the unique locator ID of each cached object along with the content group in which it was cached, and other details. To view more details of a specific cached object, use the -locator parameter along with this command.

Synopsys

show cache object [(-url <URL> (-host <string> [-port <port>] [-groupName <string>] [-httpMethod (GET | POST)])) | -locator <positive_integer> | -httpStatus <positive_integer> | -group <string> | -ignoreMarkerObjects (ON | OFF) | -includeNotReadyObjects (ON | OFF)]

Arguments

url

URL of the particular object whose details is required. Parameter "host" must be specified along with the URL.

locator

ID of the cached object.

httpStatus

HTTP status of the object.

host

Host name of the object. Parameter "url" must be specified.

port

Host port of the object. You must also set the Host parameter.

Default value: 80

Minimum value: 1

groupName

Name of the content group to which the object belongs. It will display only the objects belonging to the specified content group. You must also set the Host parameter.

httpMethod

HTTP request method that caused the object to be stored.

Possible values: GET, POST

Default value: NS_HTTP_METHOD_GET

group

Name of the content group whose objects should be listed.

ignoreMarkerObjects

Ignore marker objects. Marker objects are created when a response exceeds the maximum or minimum response size for the content group or has not yet received the minimum number of hits for the content group.

Possible values: ON, OFF

includeNotReadyObjects

Include responses that have not yet reached a minimum number of hits before being cached.

Possible values: ON, OFF

summary

fullValues

Outputs

cacheResSize

Cache response size of the object.

cacheResHdrSize

Cache response header size of the object.

cacheETag

Cache ETag of the object.

httpStatusOutput

HTTP status of the object.

cacheResLastMod

Value of "Last-modified" header.

cacheControl

Cache-Control header of the object.

cacheResDate

Value of "Date" header

contentGroup

Name of the contentgroup in which it is stored.

destIP

Destination IP.NOTE: This attribute is deprecated.This is no more in use.

destIPV46

Destination IP.

destPort

Destination Port.

cacheCellComplex

The state of the parameterized caching on this cell.

hitParams

Parameterized hit evaluation of an object.

hitValues

Values of hitparams for this object.

cacheCellReqTime

Required time of the cache cell object.

cacheCellResTime

Response time to the cache cell object.

cacheCurAge

Current age of the cache object.

cacheCellExpires

Expiry time of the cache cell object in seconds.

cacheCellExpiresMilliSec

Expiry time of the cache cell object in milliseconds.

flushed

Specifies whether the object is flushed.

prefetch

Specifies whether Integrated Cache should attempt to refresh an object immediately before it goes stale.

prefetchPeriod

The duration in seconds of the period during which prefetch should be attempted, immediately before the object's calculated expiry time.

prefetchPeriodMilliSec

The duration in milliseconds of the period during which prefetch should be attempted, immediately before the object's calculated expiry time.

cacheCellCurReaders

Current readers of the cache cell object.

cacheCellCurMisses

Current misses of the cache cell object.

cacheCellHits

Cache cell hits.

cacheCellMisses

Cache cell misses.

cacheCellDHits

Cache cell disk hits.

cacheCellGzipCompressed

The state of the response being gzip-compressed.NOTE: This attribute is deprecated.we display compression format using nsace_contenc_name

cacheCellDeflateCompressed

The state of the response being deflate-compressed.NOTE: This attribute is deprecated.we display compression format using nsace_contenc_name

cacheCellCompressionFormat

Compression format of this object. Identity means not compressed

cacheCellAppFWMetadataExists

AppFirewall cache object.

cacheCellHttp11

The state of the response to be HTTP/1.1.

cacheCellWeakEtag

The state of the weak HTTP Entity Tag in the cell.

cacheCellResBadSize

The marked state of the cell.

markerReason

Reason for marking the cell.

cacheCellPollEveryTime

The state to poll every time on object.

cacheCellEtagInserted

The state of the ETag to be inserted by IC for this object.

cacheCellReadyWithLastByte

The state of the complete arrived response.

cacheInMemory

The cache data is available in memory.

cacheInDisk

The cache data is available in disk.

cacheDirname

The directory name used if saved.

cacheFilename

The filename used if saved.

cacheCellDestipVerified

The state of DNS verification.

cacheCellFwpxyObj

The state of the object to be stored on a request to a forward proxy.

cacheCellBasefile

The state of delta being used as a basefile.

cacheCellMinHitFlag

The state of the minhit feature on this cell.

cacheCellMinHit

Min hit value for the object.

policy

Policy info for the object.

policyName

Policy which created the object.

selectorName

The hit selector for the object.

rule

Selectors for this object.

selectorValue

The HTTP request method that caused the object to be stored.

cacheUrls

List of cache object URLs.

numurls

Total number of cache object entries returned in cacheUrls field

warnBucketSkip

Bucket skipped warning.

totalObjs

Total objects.

httpCalloutCell

Is it a http callout cell ?

httpCalloutName

Name of the http callout

returnType

Return type of the http callout

httpCalloutResult

First few bytes of http callout response

ceflags

Indicates state and type of cached cell

devno

count

stateflag

---

expire cache object

Forces expiry of a cached object. You have to specify the locator ID of the cached object by using the -locator parameter.

Synopsys

expire cache object (-locator <positive_integer> | (-url <URL> (-host <string> [-port <port>] [-groupName <string>] [-httpMethod (GET | POST)])))

Arguments

locator

ID of the cached object to be expired To view the locator ID of the cached objects, use the show cache object command.

url

The URL of the object to be expired.

host

The host of the object to be expired.

port

The host port of the object to be expired.

Default value: 80

Minimum value: 1

groupName

Name of the content group to which the object belongs.

httpMethod

HTTP request method that caused the object to be stored.

Possible values: GET, POST

Default value: NS_HTTP_METHOD_GET

---

flush cache object

Removes a cached object from memory and from disk (if it has a disk copy). You have to specify the locator ID of the cached object by using the -locator parameter

Synopsys

flush cache object (-locator <positive_integer> | (-url <URL> (-host <string> [-port <port>] [-groupName <string>] [-httpMethod (GET | POST)])))

Arguments

locator

ID of the cached object. To view the locator ID of the cached objects, use the show cache object command.

url

URL of the object to be flushed. You must also set the Host parameter.

host

Host of the object to be flushed. Must provide the "url" parameter along with the host.

port

Host port of the object to be flushed. Must provide the "host" parameter along with the port.

Default value: 80

Minimum value: 1

groupName

Name of the content group to which the object belongs. Must provide the \\"host\\" parameter along with the group name.

httpMethod

HTTP request method that caused the object to be stored. All objects cached by that method will be flushed.

Possible values: GET, POST

Default value: NS_HTTP_METHOD_GET

cache parameter

Sep 22, 2015

The following operations can be performed on "cache parameter":

set | unset | show

set cache parameter

Modifies the global configuration of the integrated cache. You can modify the settings of various parameters.

Synopsys

set cache parameter [-memLimit <MBytes>] [-via <string>] [-verifyUsing <verifyUsing>] [-maxPostLen <positive_integer>] [-prefetchMaxPending <positive_integer>] [-enableBypass (YES | NO)] [-undefAction (NOCACHE | RESET)]

Arguments

memLimit

Amount of memory available for storing the cache objects. In practice, the amount of memory available for caching can be less than half the total memory of the NetScaler appliance.

via

String to include in the Via header. A Via header is inserted into all responses served from a content group if its Insert Via flag is set.

verifyUsing

Criteria for deciding whether a cached object can be served for an incoming HTTP request. Available settings function as follows:

HOSTNAME - The URL, host name, and host port values in the incoming HTTP request header must match the cache policy. The IP address and the TCP port of the destination host are not evaluated. Do not use the HOSTNAME setting unless you are certain that no rogue client can access a rogue server through the cache.

HOSTNAME_AND_IP - The URL, host name, host port in the incoming HTTP request header, and the IP address and TCP port of

the destination server, must match the cache policy.

DNS - The URL, host name and host port in the incoming HTTP request, and the TCP port must match the cache policy. The host name is used for DNS lookup of the destination server's IP address, and is compared with the set of addresses returned by the DNS lookup.

Possible values: HOSTNAME, HOSTNAME_AND_IP, DNS

maxPostLen

Maximum number of POST body bytes to consider when evaluating parameters for a content group for which you have configured hit parameters and invalidation parameters.

Default value: 4096

Maximum value: 131072

prefetchMaxPending

Maximum number of outstanding prefetches in the Integrated Cache.

enableBypass

Evaluate the request-time policies before attempting hit selection. If set to NO, an incoming request for which a matching object is found in cache storage results in a response regardless of the policy configuration.

If the request matches a policy with a NOCACHE action, the request bypasses all cache processing.

This parameter does not affect processing of requests that match any invalidation policy.

Possible values: YES, NO

undefAction

Action to take when a policy cannot be evaluated.

Possible values: NOCACHE, RESET

---

unset cache parameter

Use this command to remove cache parameter settings.Refer to the set cache parameter command for meanings of the arguments.

Synopsys

unset cache parameter [-memLimit] [-via] [-verifyUsing] [-maxPostLen] [-prefetchMaxPending] [-enableBypass] [-undefAction]

---

show cache parameter

Displays the global configuration of the Integrated Cache.

Synopsys

show cache parameter

Arguments

format

level

Outputs

memLimit

The memory limit for the Integrated Cache.

memLimitActive

Active value of the memory limit for the Integrated Cache.

maxMemLimit

The maximum value of the memory limit for the Integrated Cache.

via

The string that is inserted in the "Via" header.

verifyUsing

The criteria for deciding whether a cached object can be served for an incoming HTTP request.

maxPostLen

The maximum POST body size that the IC can accumulate.

prefetchCur

Number of current outstanding prefetches in the IC.

prefetchMaxPending

The maximum number of outstanding prefetches on the content group.

enableBypass

When this value is set to NO, an incoming request will serve a hit if a matching object is found in cache storage, regardless of the cacheability policy configuration. If set to YES, the bound request cacheability policies are evaluated before attempting any hit selection in the cache storage. If the request matches a policy with a NOCACHE action, the request will bypass all cache processing. This flag does not affect processing of requests that match any invalidation policy.

undefAction

Action to take when a policy cannot be evaluated.

enableDiskCache

The disk cache parameter. When this value is set to YES, cache objects can be saved on disk. If set to NO, objects will never be stored in disk.

cache policy

Sep 22, 2015

The following operations can be performed on "cache policy":

add | rm | set | unset | show | stat | rename

add cache policy

Creates an integrated caching policy. The newly created policy is in inactive state. To activate the policy, use the bind cache global command.

Synopsys

add cache policy <policyName> -rule <expression> -action <action> [-storeInGroup <string>] [-invalGroups <string> ...] [-invalObjects <string> ...] [-undefAction (NOCACHE | RESET)]

Arguments

policyName

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the policy is created.

rule

Expression against which the traffic is evaluated.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to content that matches the policy.

* CACHE or MAY_CACHE action - positive cachability policy

* NOCACHE or MAY_NOCACHE action - negative cachability policy

* INVAL action - Dynamic Invalidation Policy

Possible values: CACHE, NOCACHE, MAY_CACHE, MAY_NOCACHE, INVAL

storeInGroup

Name of the content group in which to store the object when the final result of policy evaluation is CACHE. The content group must exist before being mentioned here. Use the "show cache contentgroup" command to view the list of existing content groups.

invalGroups

Content group(s) to be invalidated when the INVAL action is applied. Maximum number of content groups that can be specified is 16.

invalObjects

Content groups(s) in which the objects will be invalidated if the action is INVAL.

undefAction

Action to be performed when the result of rule evaluation is undefined.

Possible values: NOCACHE, RESET

---

rm cache policy

Removes the specified caching policy. Make sure that the policy is not bound globally or to a virtual server. A bound policy cannot be removed.

Synopsys

rm cache policy <policyName>

Arguments

policyName

Name of the cache policy to be removed.

---

set cache policy

Modifies the specified attributes of an existing cache policy. The rule, flow type, can be changed only if action and undefAction (if present) are of NEUTRAL flow type.

Synopsys

set cache policy <policyName> [-rule <expression>] [-action <action>] [-storeInGroup <string>] [-invalGroups <string> ...] [-invalObjects <string> ...] [-undefAction (NOCACHE | RESET)]

Arguments

policyName

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the policy is created.

rule

Expression against which the traffic is evaluated.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to content that matches the policy.

* CACHE or MAY_CACHE action - positive cachability policy

* NOCACHE or MAY_NOCACHE action - negative cachability policy

* INVAL action - Dynamic Invalidation Policy

Possible values: CACHE, NOCACHE, MAY_CACHE, MAY_NOCACHE, INVAL

storeInGroup

Name of the content group in which to store the object when the final result of policy evaluation is CACHE. The content group must exist before being mentioned here. Use the "show cache contentgroup" command to view the list of existing content groups.

invalGroups

Content group(s) to be invalidated when the INVAL action is applied. Maximum number of content groups that can be specified is 16.

invalObjects

Content groups(s) in which the objects will be invalidated if the action is INVAL.

undefAction

Action to be performed when the result of rule evaluation is undefined.

Possible values: NOCACHE, RESET

Example

```
set cache policy pol9 -rule "http.req.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"
```

---

unset cache policy

Use this command to remove cache policy settings.Refer to the set cache policy command for meanings of the arguments.

Synopsys

unset cache policy <policyName> [-storeInGroup] [-invalGroups] [-invalObjects] [-undefAction]

---

show cache policy

Displays all configured cache policies. To display details about a particular cache policy, specify the name of the policy. When all caching policies are displayed, the order of the displayed policies within each group is the same as the evaluation order of the policies. There are three groups: request policies, response policies, and dynamic invalidation policies.

Synopsys

show cache policy [<policyName>] show cache policy stats - alias for 'stat cache policy'

Arguments

policyName

Name of the cache policy about which to display details.

summary

fullValues

format

level

Outputs

stateflag

rule

The request/response rule that will trigger the specified action.

action

The integrated cache action to be applied when the system sees content that matches the rules.

storeInGroup

The content group that will store the object when the action directive is CACHE.

invalGroups

The content group(s) to be invalidated when the action directive is INVAL.

invalObjects

The content group(s) whose objects will be invalidated when the action directive is INVAL.

priority

Priority.

hits

Hits.

undefAction

A CACHE action, to be used by the policy when the rule evaluation turns out to be undefined.

undefHits

Number of Undef hits.

flags

Flag.

precedeDefRules

Override default request/response cacheability rules.NOTE: This attribute is deprecated.Since pre-builtin, built-in and post-built-in policies are in same policy bank, this is no longer needed

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

bindPolicyType

vserverType

builtin

devno

count

---

stat cache policy

Displays a summary of cache policy statistics.

Synopsys

stat cache policy [<policyName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

policyName

Name of the cache policy for which to display statistics. If you do not set this parameter, statistics are shown for all cache policies.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat cache policy
```

---

rename cache policy

Renames an existing cache policy.

Synopsys

rename cache policy <policyName>@ <newName>@

Arguments

policyName

Existing name of the cache policy.

newName

New name for the cache policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

```
rename cache policy oldname newname
```

cache policylabel

Sep 22, 2015

The following operations can be performed on "cache policylabel":

add | rm | bind | unbind | show | stat | rename

add cache policylabel

Creates a user-defined cache policy label. A policy label is a bind point of a group of policies.

Synopsys

add cache policylabel <labelName> -evaluates <evaluates>

Arguments

labelName

Name for the label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the label is created.

evaluates

When to evaluate policies bound to this label: request-time or response-time.

Possible values: REQ, RES, MSSQL_REQ, MSSQL_RES, MYSQL_REQ, MYSQL_RES

Example

```
add cache policylabel cache_http_url -evaluates REQ
```

---

rm cache policylabel

Removes the specified integrated caching policy label.

Synopsys

rm cache policylabel <labelName>

Arguments

labelName

Name of the label to be removed.

Example

```
rm cache policylabel cache_http_url
```

---

bind cache policylabel

Binds a cache policy to a policy label.

Synopsys

bind cache policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)]

Arguments

labelName

Name of the cache policy label to which to bind the policy.

policyName

Name of the cache policy to bind to the policy label.

Example

i) bind cache policylabel cache_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind cache policylabel cache_http_url pol_2 2

---

unbind cache policylabel

Unbinds a policy from a cache-policy label.

Synopsys

unbind cache policylabel <labelName> -policyName <string> [-priority <positive_integer>]

Arguments

labelName

Name of the cache policy label from which to unbind the policy.

policyName

Name of the policy to unbind from the label.

priority

Required only if you want to unbind a NOPOLICY that might have been bound to this policy label.

Minimum value: 1

Maximum value: 2147483647

Example

unbind cache policylabel cache_http_url pol_1

---

show cache policylabel

Displays information about all cache-policy labels or about the specified cache-policy label.

Synopsys

show cache policylabel [<labelName>]

Arguments

labelName

Name of the cache-policy label about which to display information.

summary

fullValues

format

level

Outputs

stateflag

flags

evaluates

When to evaluate policies bound to this label: request-time or response-time.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the cache policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next-lower priority.

labelType

Type of policy label to invoke: an unnamed label associated with a virtual server, or user-defined policy label.

labelName

Name of the policy label to invoke if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound cache policy.

builtin

devno

count

Example

i) show cache policylabel cache_http_url ii) show cache policylabel

---

stat cache policylabel

Displays statistics of cache policy label(s).

Synopsys

stat cache policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

labelName

Name of the cache-policy label for which to display statistics. If you do not set this parameter statistics are shown for all cache-policy labels.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

---

rename cache policylabel

Renames a cache-policy label.

Synopsys

rename cache policylabel <labelName>@ <newName>@

Arguments

labelName

Existing name of the cache-policy label.

newName

New name for the cache-policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

rename cache policylabel oldname newname

---

cache selector

Sep 22, 2015

The following operations can be performed on "cache selector":

add | rm | set | show

add cache selector

Creates an Integrated Cache selector. A selector is an abstraction for a collection of PIXL expressions. After creating a selector, you can use it as a hit selector, invalidation selector, or both. You must specify at least one expression when you create a selector.

Synopsys

add cache selector <selectorName> <rule> ...

Arguments

selectorName

Name for the selector. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

rule

One or multiple PIXL expressions for evaluating an HTTP request or response.

---

rm cache selector

Removes cache selectors. Note: A selector being used as a hit or invalidation selector in any content group cannot be removed without unsetting it from the content group.

Synopsys

rm cache selector <selectorName>

Arguments

selectorName

Name of the selector.

---

set cache selector

Modify the set of PIXL expressions associated with a cache selector.

Synopsys

set cache selector <selectorName> <rule> ...

Arguments

selectorName

Name of the selector to be modified.

rule

One or multiple PIXL expressions for evaluating an HTTP request or response.

---

show cache selector

Displays all cache selectors, or the specified.

Synopsys

show cache selector [<selectorName>]

Arguments

selectorName

Name of the selector to display.

summary

fullValues

format

level

Outputs

flags

Flags.

rule

Rule.

devno

count

stateflag

---

cache stats

Sep 22, 2015

The following operations can be performed on "cache stats":

show cache stats

show cache stats is an alias for stat cache

Synopsys

show cache stats - alias for 'stat cache'

CLI Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- alias
- backup
- batch
- cli attribute
- cli mode
- cli prompt
- cls
- config
- exit
- help
- history
- man
- quit
- source
- unalias
- whoami

alias

Sep 22, 2015

The following operations can be performed on "alias":

alias

Create (short) aliases for (long) commands. Aliases are saved across NSCLI sessions. If no argument is specified, the alias command will display existing aliases.

Synopsys

alias [<pattern> [(command)]]

Arguments

pattern

Alias name. (Can be a regular expression.)

Example

```
alias info "show ns info"
```

backup

Sep 22, 2015

The following operations can be performed on "backup":

backup

backup cache object to local disk

Synopsys

backup -pattern <string>

Arguments

pattern

Name of the alias

Example

backup cache object -locator <id>

batch

Sep 22, 2015

The following operations can be performed on "batch":

batch

Use this command to read the contents of a file and execute each line as a separate CLI command. Each command in the file must be on a separate line. Lines starting with # are considered comments.

Synopsys

batch -fileName <input_filename> [-outfile <output_filename>] [-ntimes <positive_integer>]

Arguments

fileName

The name of the batch file.

outfile

The name of the file where the executed batch file will write its output. The default is standard output.

ntimes

The number of times the batch file will be executed.

Default value: 1

Example

batch -f cmds.txt

cli attribute

Sep 22, 2015

The following operations can be performed on "cli attribute":

show cli attribute

Display attributes of the NetScaler CLI

Synopsys

show cli attribute

Outputs

qquote

The construct that is used to quote strings that are to be taken as-is, without interpreting escape sequences like "

".

This construct consists of: a 'q', followed by a delimiter character; the string follows immediately after the delimiter and is terminated by the first matching delimiter character. (The set of possible delimiter characters is listed below.)

For example, q/a

/ will result in a three-character string ('a', '', 'n'); whereas "a

" results in a two-character string ('a' followed by a newline).

qquoteDelimiters

The set of characters that can be used as the delimiter in a q// construct. Characters shown in pairs must be used that way, whereas characters shown singly server as thier own matching delimiter.

For example, q?abc? and q{abc} are valid q// constructs, and evaluate to the string "abc"; q{abc{ is however not a valid q// construct so it will evaluate to the string "q{abc{".

cli mode

Sep 22, 2015

The following operations can be performed on "cli mode":

set | unset | show

set cli mode

Use this command to specify how the CLI should display command output.

Synopsys

set cli mode [-page (ON | OFF)] [-total (ON | OFF)] [-color (ON | OFF)] [-disabledFeatureAction <disabledFeatureAction>] [-timeout <secs>] [-regex (ON | OFF)]

Arguments

page

Determines whether output that spans more than one screen is "paged". Specify ON to pause the display after each screen of ouput.

Possible values: ON, OFF

Default value: OFF

total

Determines whether CLI "show" commands display a total count of objects before displaying the objects themselves.

Possible values: ON, OFF

Default value: OFF

color

Specifies whether output can be shown in color, if the terminal supports it.

Possible values: ON, OFF

Default value: OFF

disabledFeatureAction

Specifies what will happen when a configuration command is issued for a disabled feature. The following values are allowed:

NONE - The action is allowed, and no warning message is issued.;

ALLOW - The action is allowed, but a warning message is issued.;

DENY - The action is not allowed.;

HIDE - Commands that configure disabled features are hidden, and the CLI behaves as if they did not exist.

Possible values: NONE, ALLOW, DENY, HIDE

Default value: NS_ALLOW

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

Default value: 300

Maximum value: 100000000

regex

If ON, regular expressions can be used as argument values

Possible values: ON, OFF

Default value: ON

---

unset cli mode

Use this command to remove cli mode settings.Refer to the set cli mode command for meanings of the arguments.

Synopsys

unset cli mode [-page] [-total] [-color] [-disabledFeatureAction] [-timeout] [-regex]

---

show cli mode

Use this command to display the current settings of parameters that can be set with the 'set cli mode' command.

Synopsys

show cli mode

Arguments

format

level

Outputs

page

Determines whether output that spans more than one screen is "paged". Specify ON to pause the display after each screen of ouput.

total

Determines whether CLI "show" commands display a total count of objects before displaying the objects themselves.

color

Specifies whether output can be shown in color, if the terminal supports it.

disabledFeatureAction

Specifies what will happen when a configuration command is issued for a disabled feature. The following values are allowed:

NONE - The action is allowed, and no warning message is issued.;

ALLOW - The action is allowed, but a warning message is issued.;

DENY - The action is not allowed.;

HIDE - Commands that configure disabled features are hidden, and the CLI behaves as if they did not exist.

argMark

mark

noLicenseAction

no licence

diagLevel

diagnostic level

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

regex

If ON, regular expressions can be used as argument values

r

regular expression

format

format

stats

serverPort

cli prompt

Sep 22, 2015

The following operations can be performed on "cli prompt":

clear | set | show

clear cli prompt

Use this command to return the CLI prompt to the default (a single '>').

Synopsys

clear cli prompt

---

set cli prompt

Use this command to customize the CLI prompt.

Synopsys

set cli prompt <promptString>

Arguments

promptString

The prompt string. The following special values are allowed:

%! - will be replaced by the history event number

%u - will be replaced by the NetScaler user name

%h - will be replaced by the NetScaler hostname

%t - will be replaced by the current time

%T - will be replaced by the current time (24 hr format)

%d - will be replaced by the current date

%s - will be replaced by the node state

Example

```
> set cli prompt "%h %T"  Done lb-ns1 15:16>
```

---

show cli prompt

Use this command to display the current CLI prompt, with special values like '%h' unexpanded.

Synopsys

show cli prompt

Arguments

format

level

Outputs

promptString

Example

```
10.101.4.22 15:20> sh cli prompt        CLI prompt is set to "%h %T"  Done
```

cls

Sep 22, 2015

The following operations can be performed on "cls":

cls

Clear the screen and reposition cursor at top right.

Synopsys

cls

config

Sep 22, 2015

The following operations can be performed on "config":

config

Enter this command to enter contextual mode.

Synopsys

config

exit

Sep 22, 2015

The following operations can be performed on "exit":

exit

Use this command to back out one level in config mode, or to terminate the CLI when not in config mode.);

Synopsys

exit

help

Sep 22, 2015

The following operations can be performed on "help":

help

Use this command to display help information for a CLI command, for a group of commands, or for all CLI commands.

Synopsys

help [(commandName) | <groupName> | -all]

Arguments

commandName

The name of a command for which you want full usage information.

groupName

The name of a command group for which you want basic usage information.

all

Use this option to request basic usage information for all commands.

Example

1.To view help information for adding a virtual server, enter the following CLI command: help add vserver The following information is displayed: Usage: add vserver <vS

history

Sep 22, 2015

The following operations can be performed on "history":

history

Use this command to see the history of the commands executed on CLI.

Synopsys

history

Example

```
history                1 add snmp trap SPECIFIC 10.102.130.228    2 save config    3 show system session    4 swhell
```

man

Sep 22, 2015

The following operations can be performed on "man":

man

Use this command to invoke the man page for the specified command. You can specify the command in full, or partially, if it is uniquely resolvable.

Synopsys

man [(commandName)]

Arguments

commandName

The name of the command.

Example

man add vs

quit

Sep 22, 2015

The following operations can be performed on "quit":

quit

Use this command to terminate the CLI. Note: typing <Ctrl>+<d> will also terminate the CLI.

Synopsys

quit

source

Sep 22, 2015

The following operations can be performed on "source":

source

Use this command to read the contents of a file and execute each line as a separate CLI command. Each command in the file being read must be on a separate line. Lines starting with # are considered comments.

Synopsys

source <fileName>

Arguments

fileName

The name of the file to be sourced.

Example

source cmds.txt

unalias

Sep 22, 2015

The following operations can be performed on "unalias":

unalias

Remove an alias

Synopsys

unalias <pattern>

Arguments

pattern

Name of the alias

Example

unalias info

whoami

Sep 22, 2015

The following operations can be performed on "whoami":

whoami

Show the current user.

Synopsys

whoami

Outputs

userName

loggedIn

Cluster Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- cluster
- cluster files
- cluster instance
- cluster node
- cluster nodegroup
- cluster sync

cluster

Sep 22, 2015

The following operations can be performed on "cluster":

join cluster

Joins the appliance to the cluster. You must execute this command from the NetScaler IP (NSIP) address of the node that you want to add to the cluster. This command is the second part of the two-step process of adding a cluster node. The first part is adding this node to the cluster by using the add cluster node command from the cluster IP address. This operation is not permitted if any node in the cluster is in the Sync state.

Synopsys

join cluster -clip <ip_addr> {-password }

Arguments

clip

Cluster IP address to which to add the node.

password

Password for the nsroot account of the configuration coordinator (CCO).

cluster files

Sep 22, 2015

The following operations can be performed on "cluster files":

sync cluster files

Synchronizes SSL Certificates, SSL CRL lists, SSL VPN bookmarks, and other files from the configuration coordinator (CCO) to the other cluster nodes. Execute this command from the cluster IP address only. This command is automatically triggered from the CCO when a new node is added to a cluster and periodically triggered to synchronize updated files between the cluster nodes. Note: Files on non-CCO nodes are not deleted if they do no not exist on the CCO.

Synopsys

sync cluster files [<Mode> ...]

Arguments

Mode

The directories and files to be synchronized. The available settings function as follows:

Mode Paths

all /nsconfig/ssl/

/var/netscaler/ssl/

/var/vpn/bookmark/

/nsconfig/dns/

/nsconfig/htmlinjection/

/netscaler/htmlinjection/ens/

/nsconfig/monitors/

/nsconfig/nstemplates/

/nsconfig/ssh/

/nsconfig/rc.netscaler

/nsconfig/resolv.conf

/nsconfig/inetd.conf

/nsconfig/syslog.conf

/nsconfig/snmpd.conf

/nsconfig/ntp.conf

/nsconfig/httpd.conf

/nsconfig/sshd_config

/nsconfig/hosts

/nsconfig/enckey

/var/nslw.bin/etc/krb5.conf

/var/nslw.bin/etc/krb5.keytab

/var/lib/likewise/db/

/var/download/

/var/wi/tomcat/webapps/

/var/wi/tomcat/conf/Catalina/localhost/

/var/wi/java_home/lib/security/cacerts

/var/wi/java_home/jre/lib/security/cacerts

ssl /nsconfig/ssl/

/var/netscaler/ssl/

bookmarks /var/vpn/bookmark/

dns /nsconfig/dns/

htmlinjection /nsconfig/htmlinjection/

imports /var/download/

misc /nsconfig/license/

/nsconfig/rc.conf

all_plus_misc Includes *all* files and /nsconfig/license/ and /nsconfig/rc.conf.

Default value: all

Example

sync cluster files ssl or sync cluster files all

cluster instance

Sep 22, 2015

The following operations can be performed on "cluster instance":

add | rm | set | unset | enable | disable | show | stat

add cluster instance

Adds a cluster instance to the appliance. Execute this command on only the first node that you add to the cluster.

Synopsys

add cluster instance <clId> [-deadInterval <secs>] [-helloInterval <msecs>] [-preemption (ENABLED | DISABLED)]

Arguments

clId

Unique number that identifies the cluster.

Minimum value: 1

Maximum value: 16

deadInterval

Amount of time, in seconds, after which nodes that do not respond to the heartbeats are assumed to be down.

Default value: 3

Minimum value: 3

Maximum value: 60

helloInterval

Interval, in milliseconds, at which heartbeats are sent to each cluster node to check the health status.

Default value: 200

Minimum value: 200

Maximum value: 1000

preemption

Preempt a cluster node that is configured as a SPARE if an ACTIVE node becomes available.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add cluster instance 1
```

---

rm cluster instance

Removes the cluster instance from the node. You must execute this command on the NetScaler IP (NSIP) address of the node.

Synopsys

rm cluster instance <clId>

Arguments

clId

Unique number that identifies the cluster.

Minimum value: 1

Maximum value: 16

Example

```
rm cluster instance 1
```

---

set cluster instance

Modifies the specified attributes of a cluster instance.

Synopsys

set cluster instance <clId> [-deadInterval <secs>] [-helloInterval <msecs>] [-preemption (ENABLED | DISABLED)]

Arguments

clId

ID of the cluster instance to be modified.

Minimum value: 1

Maximum value: 16

deadInterval

Amount of time, in seconds, after which nodes that do not respond to the heartbeats are assumed to be down.

Default value: 3

Minimum value: 3

Maximum value: 60

helloInterval

Interval, in milliseconds, at which heartbeats are sent to each cluster node to check the health status.

Default value: 200

Minimum value: 200

Maximum value: 1000

preemption

Preempt a cluster node that is configured as a SPARE if an ACTIVE node becomes available.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set cluster instance 1 -preemption ENABLED
```

---

unset cluster instance

Use this command to remove cluster instance settings.Refer to the set cluster instance command for meanings of the arguments.

Synopsys

unset cluster instance <clId> [-deadInterval] [-helloInterval] [-preemption]

---

enable cluster instance

Enables a cluster instance.

Synopsys

enable cluster instance <clId>

Arguments

clId

ID of the cluster instance that you want to enable.

Minimum value: 1

Maximum value: 16

Example

```
enable cluster instance 1
```

---

disable cluster instance

Disables a cluster instance.

Synopsys

disable cluster instance <clId>

Arguments

clId

ID of the cluster instance that you want to disable.

Minimum value: 1

Maximum value: 16

Example

disable cluster instance 1

---

show cluster instance

Displays information about the cluster instance and its nodes.

Synopsys

show cluster instance [<clId>]

Arguments

clId

Unique number that identifies the cluster.

Minimum value: 1

Maximum value: 16

summary

fullValues

format

level

Outputs

deadInterval

Amount of time, in seconds, after which nodes that do not respond to the heartbeats are assumed to be down.

helloInterval

Interval, in milliseconds, at which heartbeats are sent to each cluster node to check the health status.

preemption

Preempt a cluster node that is configured as a SPARE if an ACTIVE node becomes available.

adminstate

Cluster Admin State.

propState

Enable/Disable the execution of commands on the cluster. This will not impact the execution of commands on individual cluster nodes by using the NSIP.

nodeId

The unique number that identiies a cluster.

IPAddress

The IP Address of the node.

flags

The flags for this entry.

masterState

Master state.

health

Node Health state.

clusterHealth

Node clusterd state.

effectiveState

Node effective health state.

state

Active, Spare or Passive. An active node serves traffic. A spare node serves as a backup for active nodes. A passive node does not serve traffic. This may be useful during temporary maintenance activity where it is desirable that the node takes part in the consensus protocol, but not serve traffic.

flag

Cluster Flag.

operationalstate

Cluster Operational State.

status

Cluster Operational State.

isConfigurationCoordinator

This argument is used to determine whether the node is configuration coordinator (CCO).

isLocalnode

This argument is used to determine whether it is local node.

RSSKeyMismatch

This argument is used to determine if there is a RSS key mismatch at cluster instance level.

LicenseMismatch

This argument is used to determine if there is a License mismatch at cluster instance level.

NodeRSSKeyMismatch

This argument is used to determine if there is a RSS key mismatch at cluster node level.

NodeLicenseMismatch

This argument is used to determine if there is a License mismatch at cluster node level.

stateflag

State Flag.

operationalPropState

Cluster Operational Propagation State.

devno

count

Example

An example of the command's output is as follows: 1)Cluster ID: 1 Dead Interval: 3 secs Hello Interval: 200 msecs Preemption:

---

stat cluster instance

Displays statistics for a cluster instance.

Synopsys

stat cluster instance [<clId>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clId

ID of the cluster instance for which to display statistics.

Minimum value: 1

Maximum value: 16

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Cluster size (CLNumNodes)

Number of nodes in the cluster.

Cluster status (CLCurEnable)

State of the cluster.

Configuration coordinator (CLViewLeader)

NSIP address of the Configuration Coordinator of the cluster.

Total steered packets (TotSteeredPkts)

Total number of packets steered on the cluster backplane.

Traffic received (Bkplane Rx)

Traffic received on backplane (in mbits)

Traffic transmitted (Bkplane Tx)

Traffic transmitted from backplane (in mbits)

Dropped steered packets (DFDdropPkts)

Number of steered packets that are dropped.

Propagation timeout (propTimeout)

Number of times the update to the client timed-out.

cluster node

Sep 22, 2015

The following operations can be performed on "cluster node":

add | set | unset | rm | show | stat

add cluster node

Adds a NetScaler appliance to a cluster.

Synopsys

add cluster node <nodeId>@ <IPAddress>@ [-state <state>] [-backplane <interface_name>@] [-priority <positive_integer>]

Arguments

nodeId

Unique number that identifies the cluster node.

Maximum value: 31

IPAddress

NetScaler IP (NSIP) address of the appliance to add to the cluster. Must be an IPv4 address.

state

Admin state of the cluster node. The available settings function as follows:

ACTIVE - The node serves traffic.

SPARE - The node does not serve traffic unless an ACTIVE node goes down.

PASSIVE - The node does not serve traffic, unless you change its state. PASSIVE state is useful during temporary maintenance activities in which you want the node to take part in the consensus protocol but not to serve traffic.

Possible values: ACTIVE, SPARE, PASSIVE

Default value: NSACL_NODEST_PASSIVE

backplane

Interface through which the node communicates with the other nodes in the cluster. Must be specified in the three-tuple form n/c/u, where n represents the node ID and c/u refers to the interface on the appliance.

Minimum value: 1

priority

Preference for selecting a node as the configuration coordinator. The node with the lowest priority value is selected as the configuration coordinator.

When the current configuration coordinator goes down, the node with the next lowest priority is made the new configuration coordinator. When the original node comes back up, it will preempt the new configuration coordinator and take over as the configuration coordinator.

Note: When priority is not configured for any of the nodes or if multiple nodes have the same priority, the cluster elects one of the nodes as the configuration coordinator.

Default value: 31

Maximum value: 31

Example

```
add cluster node 1 1.1.1.1 -backplane 1/1/1 -state ACTIVE
```

---

set cluster node

Modifies the attributes of a cluster node.

Synopsys

set cluster node <nodeId>@ [-state <state>] [-backplane <interface_name>@] [-priority <positive_integer>]

Arguments

nodeId

ID of the cluster node to be modified.

Maximum value: 31

state

Admin state of the cluster node. The available settings function as follows:

ACTIVE - The node serves traffic.

SPARE - The node does not serve traffic unless an ACTIVE node goes down.

PASSIVE - The node does not serve traffic, unless you change its state. PASSIVE state is useful during temporary maintenance activities in which you want the node to take part in the consensus protocol but not to serve traffic.

Possible values: ACTIVE, SPARE, PASSIVE

Default value: NSACL_NODEST_PASSIVE

backplane

Interface through which the node communicates with the other nodes in the cluster. Must be specified in the three-tuple form n/c/u, where n represents the node ID and c/u refers to the interface on the appliance.

Minimum value: 1

priority

Preference for selecting a node as the configuration coordinator. The node with the lowest priority value is selected as the configuration coordinator.

When the current configuration coordinator goes down, the node with the next lowest priority is made the new configuration coordinator. When the original node comes back up, it will preempt the new configuration coordinator and take over as the configuration coordinator.

Note: When priority is not configured for any of the nodes or if multiple nodes have the same priority, the cluster elects one of the nodes as the configuration coordinator.

Default value: 31

Maximum value: 31

Example

```
set cluster node 1 -state PASSIVE
```

---

unset cluster node

Use this command to remove cluster node settings.Refer to the set cluster node command for meanings of the arguments.

Synopsys

```
unset cluster node <nodeId>@ [-state] [-backplane] [-priority]
```

---

rm cluster node

Removes a node from the cluster and removes the cluster instance from the node. You must execute this command on the cluster IP address.

Synopsys

```
rm cluster node <nodeId>
```

Arguments

nodeId

ID of the cluster node to be removed from the cluster.

Maximum value: 31

Example

```
rm cluster node 1
```

---

show cluster node

Displays information about the cluster node.

Synopsys

```
show cluster node [<nodeId>@]
```

Arguments

nodeId

ID of the cluster node for which to display information. If an ID is not provided, information about all nodes is shown.

Default value: 255

Maximum value: 31

summary

fullValues

format

level

Outputs

IPAddress

The IP Address of the node.

flags

The flags for this entry.

clusterHealth

Node clusterd state.

effectiveState

Node effective health state.

operationalSyncState

Node Operational Reconciliation state.

masterState

Node Master state.

health

Node Health state.

state

Active, Spare or Passive.

backplane

Interface through which the node communicates with the other nodes in the cluster. Must be specified in the three-tuple form n/c/u, where n represents the node ID and c/u refers to the interface on the appliance.

priority

Preference for selecting a node as the configuration coordinator. The node with the lowest priority value is selected as the configuration coordinator.

When the current configuration coordinator goes down, the node with the next lowest priority is made the new configuration coordinator. When the original node comes back up, it will preempt the new configuration coordinator and take over as the configuration coordinator.

Note: When priority is not configured for any of the nodes or if multiple nodes have the same priority, the cluster elects one of the nodes as the configuration coordinator.

isConfigurationCoordinator

This argument is used to determine whether the node is configuration coordinator (CCO).

isLocalnode

This argument is used to determine whether it is local node.

NodeRSSKeyMismatch

This argument is used to determine if there is a RSS key mismatch at cluster node level.

NodeLicenseMismatch

This argument is used to determine if there is a License mismatch at cluster node level.

stateflag

nodeList

Nodelist for displaying Heartbeat not seen interfaces on a cluster node

ifacesList

Interface list corresponding to nodelist for Heartbeat not seen interfaces on a cluster node

enabledIfaces

Enabled Interfaces on a cluster node.

disabledIfaces

Disabled Interfaces on a cluster node.

partialFailIfaces

Partial Failure Interfaces on a cluster node.

hamonIfaces

Hamon Interfaces on a cluster node.

ifaces

Interfaces status on cluster node.

devno

count

Example

An example of the command's output is as follows: 1 cluster node: 1)Node ID: 1 IP: 1.1.1.1* Backplane:

---

stat cluster node

Displays statistics for a cluster node.

Synopsys

stat cluster node [<nodeId>@] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

nodeId

ID of the cluster node for which to display statistics. If an ID is not provided, statistics are shown for all nodes.

Maximum value: 31

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Sync state (Sync State)

Sync state of the cluster node.

Health

Health of the cluster node.

Node IP (NodeIP)

NSIP address of the cluster node.

Operational state (OpState)

Operational state of the cluster node.

Heartbeats transmitted (HB Sent)

Number of heartbeats sent. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Heartbeats received (HB Rcvd)

Number of heartbeats received. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Current node-node connections (NNMCurConn)

Number of connections open for node-to-node communication.

Node-node messages transmitted (NNMTotConnTx)

Number of node-to-node messages sent. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Node-node messages received (NNMTotConnRx)

Number of node-to-node messages received. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

PTP operational state (PTP State)

PTP state of the node. This state is Master for one node and Slave for the rest. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows UNKNOWN. When executed from the cluster IP address, shows all the statistics.

PTP packets transmitted (PTP Tx)

Number of PTP packets transmitted by the node. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

PTP packets received (PTP Rx)

Number of PTP packets received on the node. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Multicast/Broadcast send errors (NNMErrMsend)

Number of errors in sending node-to-node multicast/broadcast messages. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

(Health)

Health of the node in the cluster.

CH State

Health State of the node with respect to sync in the cluster.

cluster nodegroup

Sep 22, 2015

The following operations can be performed on "cluster nodegroup":

add | show | set | unset | bind | unbind | rm

add cluster nodegroup

Adds a nodegroup to the cluster. A nodegroup is a set of cluster nodes to which entities can be bound. Entities that are bound to a specific nodegroup are active on all the nodes of the group and not active on the nodes that are not part of the group.

Synopsys

add cluster nodegroup <name>@ [-strict (YES | NO)]

Arguments

name

Name of the nodegroup. The name uniquely identifies the nodegroup on the cluster.

strict

Specifies whether cluster nodes, that are not part of the nodegroup, will be used as backup for the nodegroup.

* Enabled - When one of the nodes goes down, no other cluster node is picked up to replace it. When the node comes up, it will continue being part of the nodegroup.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

Possible values: YES, NO

Default value: NO

Example

```
add cluster nodegroup ng1 -strict yes
```

---

show cluster nodegroup

Displays information about the available nodegroups.

Synopsys

show cluster nodegroup [<name>]

Arguments

name

Name of the nodegroup to be displayed. If a name is not provided, information about all nodegroups is displayed.

format

level

Outputs

node

Nodes in the nodegroup

strict

Specifies whether cluster nodes, that are not part of the nodegroup, will be used as backup for the nodegroup.

* Enabled - When one of the nodes goes down, no other cluster node is picked up to replace it. When the node comes up, it will continue being part of the nodegroup.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

vServer

vserver that need to be bound to this nodegroup.

currentNodeMask

Bitmap of current nodes in this nodegroup

backupNodeMask

Bitmap of backup nodes in this nodegroup

boundedEntitiesCntFromPE

Count of bounded entities to this nodegroup accoding to PE

activeList

Active node list of this nodegroup

backupList

Backup node list of this nodegroup

identifierName

stream identifier and rate limit identifier that need to be bound to this nodegroup.

stateflag

devno

count

---

set cluster nodegroup

Modifies the attributes of a cluster nodegroup.

Synopsys

set cluster nodegroup <name>@ [-strict (YES | NO)]

Arguments

name

Name of the nodegroup to be modified.

strict

Specifies whether cluster nodes, that are not part of the nodegroup, will be used as backup for the nodegroup.

* Enabled - When one of the nodes goes down, no other cluster node is picked up to replace it. When the node comes up, it will continue being part of the nodegroup.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

Possible values: YES, NO

Default value: NO

Example

```
set cluster nodegroup ng1 -strict yes
```

---

unset cluster nodegroup

Unset nodes from the given nodegroup or unset strict option.Refer to the set cluster nodegroup command for meanings of the arguments.

Synopsys

unset cluster nodegroup <name>@ [-strict]

Example

```
unset cluster nodegroup ng1 -strict
```

---

bind cluster nodegroup

Binds a cluster node or an entity to the given nodegroup. A node can be bound to more than one nodegroup.

Synopsys

bind cluster nodegroup <name> (-node <positive_integer>@ | -vServer <string> | -identifierName <string>)

Arguments

name

Name of the nodegroup to which you want to bind a cluster node or an entity.

node

ID of the node to be bound to the nodegroup.

Default value: VAL_NOT_SET

Maximum value: 31

vServer

Name of the virtual server to be bound to the nodegroup.

identifierName

Name of stream or limit identifier to be bound to the nodegroup.

Example

```
bind cluster nodegroup ng1 -vserver v1
```

---

unbind cluster nodegroup

Unbinds a cluster node or an entity from a given nodegroup.

Synopsys

unbind cluster nodegroup <name> (-node <positive_integer>@ | -vServer <string> | -identifierName <string>)

Arguments

name

Name of the nodegroup from which you want to unbind a cluster node or an entity.

node

ID of the node to be unbound from the nodegroup.

Default value: VAL_NOT_SET

Maximum value: 31

vServer

Name of the virtual server to be unbound from the nodegroup.

identifierName

Name of stream or limit identifier to be unbound from the nodegroup.

Example

```
unbind cluster nodegroup ng1 -vserver v1
```

---

rm cluster nodegroup

Removes a nodegroup from the cluster.

Synopsys

```
rm cluster nodegroup <name>@
```

Arguments

name

Name of the nodegroup to be removed.

Example

```
rm cluster nodegroup ng1
```

---

cluster sync

Sep 22, 2015

The following operations can be performed on "cluster sync":

force cluster sync

Synchronize the configurations of a cluster node from the configuration coordinator (CCO). This command must be executed from the NSIP of the node that is to be synchronized.

Synopsys

force cluster sync

Example

force cluster sync

Compression Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- cmp
- cmp action
- cmp global
- cmp parameter
- cmp policy
- cmp policylabel
- cmp stats

cmp

Sep 22, 2015

The following operations can be performed on "cmp":

Display compression statistics.

stat cmp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Bandwidth saving (%) (DlBndSav)

Bandwidth saving from delta compression expressed as percentage.

Delta compression ratio (DlCmpRt)

Ratio of compressible data received to compressed data transmitted.If this ratio is one (uncmp:1.0) that means compression is disabled or we are not able to compress even a single compressible packet.

Decompression ratio (DTCmpRt)

Ratio of decompressed data transmitted to compressed data received.

Bandwidth saving (%) (DBndSav)

Bandwidth saving from TCP compression expressed as percentage.

TCP compression ratio (TCmpRt)

Ratio of compressible data received to compressed data transmitted.If this ratio is one (uncmp:1.0) that means compression is disabled or we are not able to compress even a single compressible packet.

TCP Bandwidth saving (%) (BndSav)

Bandwidth saving from TCP compression expressed as percentage.

Total HTTP compression ratio

Ratio of total HTTP data received to total HTTP data transmitted.

HTTP Bandwidth saving (%) (HttpBndSav)

Bandwidth saving from TCP compression expressed as percentage.

HTTP compression ratio

Ratio of the compressible data received from the server to the compressed data sent to the client.

HTTP compression requests

Number of HTTP compression requests the NetScaler receives for which the response is successfully compressed. For example, after you enable compression and configure services, if you send requests to the NetScaler with the following header information: ?Accept-Encoding: gzip, deflate?, and NetScaler compresses the corresponding response, this counter is incremented.

Compressible bytes received

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Compressed bytes transmitted

Number of bytes the NetScaler sends to the client after compressing the response from the server.

Compressible packets received

Number of HTTP packets that can be compressed, which the NetScaler receives from the server.

Compressed packets transmitted

Number of HTTP packets that the NetScaler sends to the client after compressing the response from the server.

Compressible bytes received (TCmpRxB)

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Compressible packets received (TCmpRxP)

Total number of compressible packets received by NetScaler.

Compressed bytes transmitted (TCmpTxB)

Number of bytes that the NetScaler sends to the client after compressing the response from the server.

Compressed packets transmitted (TCmpTxP)

Number of TCP packets that the NetScaler sends to the client after compressing the response from the server.

Quantum compression (TCmpQuan)

Number of times the NetScaler compresses a quantum of data. NetScaler buffers the data received from the server till it reaches the quantum size and then compresses the buffered data and transmits to the client.

Push flag compression (TCmpPush)

Number of times the NetScaler compresses data on receiving a TCP PUSH flag from the server. The PUSH flag

ensures that data is compressed immediately without waiting for the buffered data size to reach the quantum size.

End Of Input compression (TCmpEoi)

Number of times the NetScaler compresses data on receiving End Of Input (FIN packet). When the NetScaler receives End Of Input (FIN packet), it compresses the buffered data immediately without waiting for the buffered data size to reach the quantum size.

Timer compression (TCmpTmr)

Number of times the NetScaler compresses data on expiration of data accumulation timer. The timer expires if the server response is very slow and consequently, the NetScaler does not receive response for a certain amount of time. Under such a condition, the NetScaler compresses the buffered data immediately without waiting for the buffered data size to reach the quantum size.

Compressed bytes received (DCmpTRxB)

Total number of compressed bytes received by NetScaler.

Compressed packets received (DCmpTRxP)

Total number of compressed packets received by NetScaler.

Decompressed bytes transmitted (DCmpTTxB)

Total number of decompressed bytes transmitted by NetScaler.

Decompressed packets transmitted (DCmpTTxP)

Total number of decompressed packets transmitted by NetScaler.

Wrong data (DCmpErrD)

Number of data errors encountered while decompressing.

Less Data (DCmpErrL)

Number of times NetScaler received less data than declared by protocol.

More Data (DCmpErrM)

Number of times NetScaler received more data than declared by protocol.

Memory failures (DCmpMem)

Number of times memory failures occurred while decompressing.

Unknown (DCmpErrU)

Number of times unknown errors occurred while decompressing.

Delta compression requests (DlCmpRx)

Total number of delta compression requests received by NetScaler.

Delta compression applied (DlDone)

Total number of delta compressions done by NetScaler.

Compressible bytes received (DlCmpRxB)

Total number of delta-compressible bytes received by NetScaler.

Compressed bytes transmitted (DlCmpTxB)

Total number of delta-compressed bytes transmitted by NetScaler.

First-time access (DlCmpFAc)

Total number of delta compression first accesses.

Compressible packets received (DlCmpRxP)

Number of delta-compressible packets received.

Compressed packets transmitted (DlCmpTxP)

Total number of delta-compressed packets transmitted by NetScaler.

Basefile requests served (DlCBSrv)

Total number of basefile requests served by NetScaler.

Basefile bytes transmitted (DlCBTxB)

Number of basefile bytes transmitted by NetScaler.

Delta compression bypassed (DlCmpEBy)

Number of times delta-compression bypassed by NetScaler.

Basefile write header failed (DlCmpEBW)

Number of times basefile could not be updated in NetScaler cache.

Basefile no-store miss (DlCmpENM)

Number of times basefile was not found in NetScaler cache.

Request information too big (DlCmpERB)

Number of times basefile request URL was too large.

Request info alloc failed (DlCmpERF)

Number of times requested basefile could not be allocated.

Session allocation failed (DlCmpESF)

Number of times delta compression session could not be allocated.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

cmp action

Sep 22, 2015

The following operations can be performed on "cmp action":

add | rm | show | rename

Creates a compression action. Note: User-defined compression actions supplement the built-in compression actions. The built-in compression actions, NOCOMPRESS, COMPRESS, GZIP, and DEFLATE, are always available. Available settings functions as follows: * NOCOMPRESS - Disables compression for data that matches the associated policy. * COMPRESS - Enable GZIP or DEFLATE compression, depending on which is supported by the browser. * GZIP - Enable GZIP compression. For browsers that do not support GZIP, compression is disabled. * DEFLATE - Enable DEFLATE compression for a specific policy. For browsers that do not support DEFLATE, compression is disabled.

add cmp action <name> <cmpType>

name

Name of the compression action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp action" or 'my cmp action').

cmpType

Type of compression performed by this action.

Available settings function as follows:

* COMPRESS - Apply GZIP or DEFLATE compression to the response, depending on the request header. Prefer GZIP.

* GZIP - Apply GZIP compression.

* DEFLATE - Apply DEFLATE compression.

* NOCOMPRESS - Do not compress the response if the request matches a policy that uses this action.

Possible values: compress, gzip, deflate, nocompress

deltaType

The type of delta action (if delta type compression action is defined).

Possible values: PERURL, PERPOLICY

Default value: NS_ACT_CMP_DELTA_TYPE_PERURL

```
add cmp action nocmp NOCOMPRESS
```

Removes the specified compression action.

rm cmp action <name>

name

Name of the action to be removed.

```
rm cmp action cmp_action_name
```

Displays information about all the built-in and user-defined compression actions, or detailed information about the specified action.

show cmp action [<name>]

name

Name of the action for which to display detailed information.

summary

fullValues

format

level

cmpType

Type of compression performed by this action.

Available settings function as follows:

* COMPRESS - Apply GZIP or DEFLATE compression to the response, depending on the request header. Prefer GZIP.

* GZIP - Apply GZIP compression.

* DEFLATE - Apply DEFLATE compression.

* NOCOMPRESS - Do not compress the response if the request matches a policy that uses this action.

deltaType

The type of delta action if compression type is delta compression.NOTE: This attribute is deprecated.Deprecating delta action in cmp policies

stateflag

flags

builtin

Flag to determine whether compression is default or not

isDefault

A value of true is returned if it is a default policy

devno

count

Example 1 The following example shows output from the show cmp action command when no custom cmp actions have been de

---

Renames a compression action.

rename cmp action <name>@ <newName>@

name

Existing name of the action.

newName

New name for the compression action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

Choose a name that can be correlated with the function that the action performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp action" or 'my cmp action').

rename cmp policy oldname newname

---

cmp global

Sep 22, 2015

The following operations can be performed on "cmp global":

bind | unbind | show

Binds (activates) the compression policy globally. Note that the compression feature requires a compression license. When you enable the compression feature, all of the built-in compression policies are bound globally.

bind cmp global <policyName> [-priority <positive_integer>] [-state (ENABLED | DISABLED)] [-gotoPriorityExpression <expression>] [-type <type>] [-invoke (<labelType> <labelName>)]

policyName

Name of the policy to bind globally.

add cmp policy pdf_cmp -rule "RES.HTTP.HEADER Content-Type CONTAINS application/pdf" -resAction COMPRESS After creating the above compression policy, you mus

---

Deactivates a globally bound HTTP compression policy.

unbind cmp global <policyName> [-type <type> [-priority <positive_integer>]]

policyName

Name of the compression policy to unbind.

To view the globally active compression policies, enter the following command: > show cmp global 5 Globally Active Compression Policies: 1) Policy Name: ns_cm

---

Displays the globally bound HTTP compression policies.

show cmp global [-type <type>]

type

Bind point to which the policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

summary

fullValues

format

level

stateflag

policyName

The name of the globally bound HTTP compression policy.

priority

Positive integer specifying the priority of the policy. The lower the number, the higher the priority. By default, polices within a label are evaluated in the order of their priority numbers.

In the configuration utility, you can click the Priority field and edit the priority level or drag the entry to a new position in the list. If you drag the entry to a new position, the priority level is updated automatically.

state

The current state of the policy binding. This attribute is relevant only for CLASSIC policies.

numpol

The number of policies bound to the bindpoint.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

policyType

Policy type (Classic/Advanced) to be bound.Used for display.

devno

count

> show cmp global 4 Globally Active Compression Policies: 1) Policy Name: ns_cmp_content_type Priority: 0 2) Policy

cmp parameter

Sep 22, 2015

The following operations can be performed on "cmp parameter":

set | unset | show

Configures the compression parameters.

set cmp parameter [-cmpLevel <cmpLevel>] [-quantumSize <positive_integer>] [-serverCmp (ON | OFF)] [-minResSize <positive_integer>] [-cmpBypassPct <positive_integer>] [-cmpOnPush (ENABLED | DISABLED)] [-policyType (CLASSIC | ADVANCED)] [-addVaryHeader (ENABLED | DISABLED)] [-externalCache (YES | NO)]

cmpLevel

Specify a compression level. Available settings function as follows:

* Optimal - Corresponds to a gzip GZIP level of 5-7.

* Best speed - Corresponds to a gzip level of 1.

* Best compression - Corresponds to a gzip level of 9.

Possible values: optimal, bestspeed, bestcompression, spdy

Default value: NSCMPLVL_OPTIMAL

quantumSize

Minimum quantum of data to be filled before compression begins.

Default value: 57344

Minimum value: 8

Maximum value: 63488

serverCmp

Allow the server to send compressed data to the NetScaler appliance. With the default setting, the NetScaler appliance handles all compression.

Possible values: ON, OFF

Default value: ON

heurExpiry

Heuristic basefile expiry.

Possible values: ON, OFF

Default value: OFF

heurExpiryThres

Threshold compression ratio for heuristic basefile expiry, multiplied by 100. For example, to set the threshold ratio to 1.25, specify 125.

Default value: 100

Minimum value: 1

Maximum value: 1000

heurExpiryHistWt

For heuristic basefile expiry, weightage to be given to historical delta compression ratio, specified as percentage. For example, to give 25% weightage to historical ratio (and therefore 75% weightage to the ratio for current delta compression transaction), specify 25.

Default value: 50

Minimum value: 1

Maximum value: 100

minResSize

Smallest response size, in bytes, to be compressed.

cmpBypassPct

NetScaler CPU threshold after which compression is not performed. Range: 0 - 100

Default value: 100

Maximum value: 100

cmpOnPush

NetScaler appliance does not wait for the quantum to be filled before starting to compress data. Upon receipt of a packet with a PUSH flag, the appliance immediately begins compression of the accumulated packets.

Possible values: ENABLED, DISABLED

Default value: DISABLED

policyType

Type of policy. Available settings function as follows:

* Classic - Classic policies evaluate basic characteristics of traffic and other data.

* Advanced - Advanced policies (which have been renamed as default syntax policies) can perform the same type of evaluations as classic policies. They also enable you to analyze more data (for example, the body of an HTTP request) and to configure more operations in the policy rule (for example, transforming data in the body of a request into an HTTP header).

Possible values: CLASSIC, ADVANCED

Default value: NS_EXPR_TYPE_CLASSIC

addVaryHeader

Add the Vary header to HTTP responses being compressed. To HTTP 1.1 responses, add a Vary: User-Agent, Accept-Encoding header. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

Possible values: ENABLED, DISABLED

Default value: DISABLED

externalCache

Enable insertion of Cache-Control: private response directive to indicate response message is intended for a single user and must not be cached by a shared or proxy cache.

Possible values: YES, NO

Default value: NO

```
set cmp param -cmpLevel bestspeed -quantumSize 20480
```

Use this command to remove cmp parameter settings.Refer to the set cmp parameter command for meanings of the arguments.

```
unset cmp parameter [-cmpLevel] [-quantumSize] [-serverCmp] [-minResSize] [-cmpBypassPct] [-cmpOnPush] [-policyType] [-addVaryHeader] [-externalCache]
```

Displays the values of the compression parameters. Example: > show cmp parameter Configured compression parameters: Compression level: optimal Quantum size: 4555 Server-side compression: ON Minimum HTTP response size for compression: 0 CPU load at which to bypass compression: 100% Compression on PUSH: DISABLED Compression policy type: CLASSIC Vary header insertion: DISABLED Disable external cache: NO

show cmp parameter

format

level

cmpLevel

Specify a compression level. Available settings function as follows:

* Optimal - Corresponds to a gzip GZIP level of 5-7.

* Best speed - Corresponds to a gzip level of 1.

* Best compression - Corresponds to a gzip level of 9.

quantumSize

Minimum quantum of data to be filled before compression begins.

serverCmp

Compression enabled/disabled at back-end server.

heurExpiry

Heuristic basefile expiry.NOTE: This attribute is deprecated.Deprecating delta action in cmp policies

heurExpiryThres

Threshold compression ratio for heuristic basefile expiry, multiplied by 100. For example, to set the threshold ratio to 1.25, specify 125.NOTE: This attribute is deprecated.Deprecating delta action in cmp policies

heurExpiryHistWt

For heuristic basefile expiry, weightage to be given to historical delta compression ratio, specified as percentage. For example, to give 25% weightage to historical ratio (and therefore 75% weightage to the ratio for current delta compression transaction), specify 25.NOTE: This attribute is deprecated.Deprecating delta action in cmp policies

minResSize

Smallest response size, in bytes, to be compressed.

cmpBypassPct

NetScaler CPU threshold after which compression is not performed. Range: 0 - 100

cmpOnPush

NetScaler appliance does not wait for the quantum to be filled before starting to compress data. Upon receipt of a packet with a PUSH flag, the appliance immediately begins compression of the accumulated packets.

policyType

Type of policy. Available settings function as follows:

* Classic - Classic policies evaluate basic characteristics of traffic and other data.

* Advanced - Advanced policies (which have been renamed as default syntax policies) can perform the same type of evaluations as classic policies. They also enable you to analyze more data (for example, the body of an HTTP request) and to configure more operations in the policy rule (for example, transforming data in the body of a request into an HTTP header).

addVaryHeader

Add the Vary header to HTTP responses being compressed. To HTTP 1.1 responses, add a Vary: User-Agent, Accept-Encoding header. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

externalCache

Enable insertion of Cache-Control: private response directive to indicate response message is intended for a single user and must not be cached by a shared or proxy cache.

cmp policy

Sep 22, 2015

The following operations can be performed on "cmp policy":

add | rm | set | show | stat | rename

Creates a classic or default syntax HTTP compression policy. When the policy matches an HTTP request or response, the action specified in the policy is performed on the transaction. The policy can be bound globally or to an entity. For the policy to have an effect, compression must be enabled on the service.

add cmp policy <name> -rule <expression> -resAction <string>

name

Name of the HTTP compression policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Can be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policy" or 'my cmp policy').

rule

Expression that determines which HTTP requests or responses match the compression policy. Can be a classic expression or a default-syntax expression.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

resAction

The built-in or user-defined compression action to apply to the response when the policy matches a request or response.

Example 1: add cmp policy pdf_cmp -rule "RES.HTTP.HEADER Content-Type CONTAINS application/pdf" -resAction COMPRESS After creating the above compression poli

---

Removes a user-defined HTTP compression policy.

rm cmp policy <name>

name

Name of the HTTP compression policy to be removed.

rm cmp policy cmp_policy_name The "show cmp policy" command shows all currently defined HTTP compression policies.

Modifies the specified parameters of an HTTP compression policy. Note: Use the show cmp policy command to view all configured HTTP compression policies.

set cmp policy <name> [-rule <expression>] [-resAction <string>]

name

Name of the HTTP compression policy to be modified.

rule

New rule to be associated with the HTTP compression policy. You can modify the existing rule or create a new rule.

resAction

The built-in or user-defined compression action to be associated with the policy.

Example 1: add cmp policy pdf_cmp -rule "RES.HTTP.HEADER Content-Type CONTAINS application/pdf" -resAction COMPRESS After creating the above compression poli

---

Displays details of all HTTP compression policies.

show cmp policy [<name>] show cmp policy stats - alias for 'stat cmp policy'

name

Name of the HTTP compression policy for which to display details.

summary

fullValues

format

level

stateflag

expressionType

Type of policy (Classic/Advanced)

rule

The request/response rule that will trigger the specified compression action.

reqAction

The compression action to be performed on requests.

resAction

The compression action to be performed on responses.

hits

Number of hits.

txbytes

Number of bytes transferred.

rxbytes

Number of bytes received.

clientTTLB

Total client TTLB value.

clientTransactions

Number of client transactions.

serverTTLB

Total server TTLB value.

serverTransactions

Number of server transactions.

piHits

Number of hits.

piTxBytes

Number of bytes transferred.

piRxBytes

Number of bytes received.

piCltTTLB

Total client TTLB value.

piCltTransactions

Number of client transactions.

piSvrTTLB

Total server TTLB value.

piSvrTransactions

Number of server transactions.

boundTo

The name of the entity to which the policy is bound.

activePolicy

priority

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

bindPolicyType

policyType

vserverType

builtin

Flag to determine if compression policy is builtin or not

isDefault

A value of true is returned if it is a default policy

devno

count

```
> show cmp policy      4 Compression policies: 1)    Name: ns_cmp_content_type     Rule: ns_c
```

Displays compression statistics for all advanced compression policies, or for only the specified policy.

```
stat cmp policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-
clearstats ( basic | full )]
```

name

Name of the advanced compression policy for which to display statistics. If no name is specified, statistics for all advanced compression polices are shown.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

stat cmp policy

Renames a compression policy.

rename cmp policy <name>@ <newName>@

name

Existing name of the policy.

newName

New name for the compression policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Choose a name that reflects the function that the policy performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policy" or 'my cmp policy').

rename cmp policy oldname newname

---

cmp policylabel

Sep 22, 2015

The following operations can be performed on "cmp policylabel":

add | rm | bind | unbind | show | stat | rename

Creates a user-defined HTTP compression policy label for default-syntax policies. Policies that you bind to the label are evaluated only if you call the label from another policy.

add cmp policylabel <labelName> -type (REQ | RES)

labelName

Name of the HTTP compression policy label. Must begin with a letter, number, or the underscore character (_). Additional characters allowed, after the first character, are the hyphen (-), period (.) pound sign (#), space ( ), at sign (@), equals (=), and colon (:). The name must be unique within the list of policy labels for compression policies. Can be renamed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policylabel" or 'my cmp policylabel').

type

Type of packets (request packets or response) against which to match the policies bound to this policy label.

Possible values: REQ, RES

```
add cmp policylabel cmp_pol_label -type REQ
```

Removes an HTTP compression policy label.

rm cmp policylabel <labelName>

labelName

Name of the HTTP compression policy label to be removed.

```
rm cmp policylabel cmp_pol_label
```

Binds a default-syntax HTTP compression policy to an HTTP compression policy label.

```
bind cmp policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>) ]
```

labelName

Name of the HTTP compression policy label to which to bind the policy.

policyName

Name of the compression policy to bind to the label.

```
bind cmp policylabel cmp_pol_label -policyName cmp_pol -priority 1
```

Unbinds a default-syntax HTTP compression policy from an HTTP compression policy label.

```
unbind cmp policylabel <labelName> <policyName> [-priority <positive_integer>]
```

labelName

Name of the HTTP compression policy label from which to unbind the policy.

policyName

Name of the HTTP compression policy to unbind from the policy label.

priority

Priority of the NOPOLICY to unbind. Required only to unbind a NOPOLICY, if it has been bound to this policy label.

Minimum value: 1

Maximum value: 2147483647

```
unbind cmp policylabel cmp_pol_label cmp_pol
```

Displays details of configured HTTP compression policy labels.

```
show cmp policylabel [<labelName>]
```

labelName

Name of the HTTP compression policy label for which to display details.

summary

fullValues

format

level

stateflag

type

Type of packets (request packets or response) against which to match the policies bound to this policy label.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

The compression policy name.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next higher priority number in the original label.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

flowType

Flowtype of the bound compression policy.

description

Description of the policylabel

flags

devno

count

i) show cmp policylabel cmp_pol_label ii) show cmp policylabel

---

Displays statistics for all compression policy labels.

stat cmp policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

labelName

Name of the compression policy label for which to display statistics. If not specified, statistics are displayed for all compression policy labels.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy Label Hits (Hits)

Number of times policy label was invoked.

Renames a compression policylabel.

rename cmp policylabel <labelName>@ <newName>@

labelName

Existing name of the policy label.

newName

New name for the compression policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policylabel" or 'my cmp policylabel').

rename cmp policylabel oldname newname

---

cmp stats

Sep 22, 2015

The following operations can be performed on "cmp stats":

show cmp stats is an alias for stat cmp Displays compression statistics.

```
show cmp stats - alias for 'stat cmp'
```

CO Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- co parameter
- co policy

co parameter

Sep 22, 2015

The following operations can be performed on "co parameter":

Use this command to remove co parameter settings.Refer to the set co parameter command for meanings of the arguments.

unset co parameter [-cacheMaxage] [-imgType] [-JpegQualityPercent] [-inlineCssThresSize] [-inlineJsThresSize] [-inlineImgThresSize]

co policy

Sep 22, 2015

The following operations can be performed on "co policy":

Use this command to remove co policy settings.Refer to the set co policy command for meanings of the arguments.

```
unset co policy <name> [-rule] [-action]
```

Cache Redirection Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- cr policy
- cr vserver

cr policy

Sep 22, 2015

The following operations can be performed on "cr policy":

add | rm | set | show

Creates a cache redirection policy. To associate the new policy with a cache redirection virtual server, use the bind cr vserver command.

add cr policy <policyName> -rule <expression>

policyName

Name for the cache redirection policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my policy? or ?my policy?).

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic syntax.

Note:Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

builtin

Removes a cache redirection policy. You can delete a user-defined cache redirection policy that is not bound to a cache

redirection virtual server. If the policy is bound to a virtual server, you must first unbind the policy, and then remove it.

rm cr policy <policyName>

policyName

Name of the cache redirection policy to remove.

Changes the specified parameters of an existing cache redirection policy.

set cr policy <policyName> -rule <expression>

policyName

Name of the cache redirection policy to change.

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator.

For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Displays all existing cache redirection policies, or just the specified policy.

show cr policy [<policyName>]

policyName

Name of the cache redirection policy to display. If this parameter is omitted, details of all the policies are displayed.

summary

fullValues

format

level

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic syntax.

Note:Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

domain

Domain name.

vstype

Virtual server type.

csPolicyType

Indicates whether policy is PI or not.(used only during display)

builtin

devno

count

stateflag

cr vserver

Sep 22, 2015

The following operations can be performed on "cr vserver":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

Creates a cache redirection virtual server.

add cr vserver <name> [-td <positive_integer>] <serviceType> [<IPAddress> <port> [-range <positive_integer>]] [-cacheType <cacheType>] [-redirect <redirect>] [-onPolicyMatch (CACHE | ORIGIN)] [-redirectURL <URL>] [-cltTimeout <secs>] [-precedence (RULE | URL)] [-arp (ON | OFF)] [-map (ON | OFF)] [-format (ON | OFF)] [-via (ON | OFF)] [-dnsVserverName <string>] [-destinationVServer <string>] [-domain <string>] [-soPersistenceTimeOut <positive_integer>] [-soThreshold <positive_integer>] [-reuse (ON | OFF)] [-state (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-backupVServer <string>] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-l2Conn (ON | OFF)] [-backendssl (ENABLED | DISABLED)] [-Listenpolicy <expression> [-Listenpriority <positive_integer>]] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-srcIPExpr <expression>] [-originUSIP (ON | OFF)] [-usePortRange (ON | OFF)] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-icmpVsrResponse (PASSIVE | ACTIVE)]

name

Name for the cache redirection virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the cache redirection virtual server is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my server? or ?my server?).

td

Traffic Domain ID

Maximum value: 4094

serviceType

Protocol (type of service) handled by the virtual server.

Possible values: HTTP, SSL, NNTP

IPAddress

IPv4 or IPv6 address of the cache redirection virtual server. Usually a public IP address. Clients send connection

requests to this IP address.

Note: For a transparent cache redirection virtual server, use an asterisk (*) to specify a wildcard virtual server address.

cacheType

Mode of operation for the cache redirection virtual server. Available settings function as follows:

* TRANSPARENT - Intercept all traffic flowing to the appliance and apply cache redirection policies to determine whether content should be served from the cache or from the origin server.

* FORWARD - Resolve the hostname of the incoming request, by using a DNS server, and forward requests for non-cacheable content to the resolved origin servers. Cacheable requests are sent to the configured cache servers.

* REVERSE - Configure reverse proxy caches for specific origin servers. Incoming traffic directed to the reverse proxy can either be served from a cache server or be sent to the origin server with or without modification to the URL.

Possible values: TRANSPARENT, REVERSE, FORWARD

Default value: CRD_TRANSPARENT

redirect

Type of cache server to which to redirect HTTP requests. Available settings function as follows:

* CACHE - Direct all requests to the cache.

* POLICY - Apply the cache redirection policy to determine whether the request should be directed to the cache or to the origin.

* ORIGIN - Direct all requests to the origin server.

Possible values: CACHE, POLICY, ORIGIN

Default value: CRD_POLICY

onPolicyMatch

Redirect requests that match the policy to either the cache or the origin server, as specified.

Note: For this option to work, you must set the cache redirection type to POLICY.

Possible values: CACHE, ORIGIN

Default value: CRD_ORIGIN

redirectURL

URL of the server to which to redirect traffic if the cache redirection virtual server configured on the NetScaler appliance becomes unavailable.

cltTimeout

Time-out value, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

precedence

Type of policy (URL or RULE) that takes precedence on the cache redirection virtual server. Applies only to cache redirection virtual servers that have both URL and RULE based policies. If you specify URL, URL based policies are applied first, in the following order:

1. Domain and exact URL

2. Domain, prefix and suffix

3. Domain and suffix

4. Domain and prefix

5. Domain only

6. Exact URL

7. Prefix and suffix

8. Suffix only

9. Prefix only

10. Default

If you specify RULE, the rule based policies are applied before URL based policies are applied.

Possible values: RULE, URL

Default value: CS_PRIORITY_RULE

arp

Use ARP to determine the destination MAC address.

Possible values: ON, OFF

ghost

map

Obsolete.

Possible values: ON, OFF

format

via

Insert a via header in each HTTP request. In the case of a cache miss, the request is redirected from the cache server to the origin server. This header indicates whether the request is being sent

from a cache server.

Possible values: ON, OFF

Default value: ON

cacheVserver

Name of the default cache virtual server to which to redirect requests (the default target of the cache redirection virtual server).

dnsVserverName

Name of the DNS virtual server that resolves domain names arriving at the forward proxy virtual server.

Note: This parameter applies only to forward proxy virtual servers, not reverse or transparent.

destinationVServer

Destination virtual server for a transparent or forward proxy cache redirection virtual server.

domain

Default domain for reverse proxies. Domains are configured to direct an incoming request from a specified source domain to a specified target domain. There can be several configured pairs of source and target domains. You can select one pair to be the default. If the host header or URL of an incoming request does not include a source domain, this option sends the request to the specified target domain.

soPersistenceTimeOut

Time-out, in minutes, for spillover persistence.

Minimum value: 2

Maximum value: 24

soThreshold

For CONNECTION (or) DYNAMICCONNECTION spillover, the number of connections above which the virtual server enters spillover mode. For BANDWIDTH spillover, the amount of incoming and outgoing traffic (in Kbps) before spillover. For HEALTH spillover, the percentage of active services (by weight) below which spillover occurs.

Minimum value: 1

reuse

Reuse TCP connections to the origin server across client connections. Do not set this parameter unless the Service Type parameter is set to HTTP. If you set this parameter to OFF, the possible settings of the Redirect parameter function as follows:

* CACHE - TCP connections to the cache servers are not reused.

* ORIGIN - TCP connections to the origin servers are not reused.

* POLICY - TCP connections to the origin servers are not reused.

If you set the Reuse parameter to ON, connections to origin servers and connections to cache servers are reused.

Possible values: ON, OFF

Default value: ON

state

Initial state of the cache redirection virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

downStateFlush

Perform delayed cleanup of connections to this virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

backupVServer

Name of the backup virtual server to which traffic is forwarded if the active server becomes unavailable.

disablePrimaryOnDown

Continue sending traffic to a backup virtual server even after the primary virtual server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

l2Conn

Use L2 parameters, such as MAC, VLAN, and channel to identify a connection.

Possible values: ON, OFF

backendssl

Decides whether the backend connection made by NS to the origin server will be HTTP or SSL. Applicable only for SSL type CR Forward proxy vserver.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Listenpolicy

String specifying the listen policy for the cache redirection virtual server. Can be either an in-line expression or the name of a named expression.

Default value: "none"

Listenpriority

Priority of the listen policy specified by the Listen Policy parameter. The lower the number, higher the priority.

Default value: 101

Maximum value: 100

tcpProfileName

Name of the profile containing TCP configuration information for the cache redirection virtual server.

httpProfileName

Name of the profile containing HTTP configuration information for cache redirection virtual server.

comment

Comments associated with this virtual server.

srcIPExpr

Expression used to extract the source IP addresses from the requests originating from the cache. Can be either an in-line expression or the name of a named expression.

originUSIP

Use the client?s IP address as the source IP address in requests sent to the origin server.

Note: You can enable this parameter to implement fully transparent CR deployment.

Possible values: ON, OFF

Default value: OFF

usePortRange

Use a port number from the port range (set by using the set ns param command, or in the Create Virtual Server (Cache Redirection) dialog box) as the source port in the requests sent to the origin server.

Possible values: ON, OFF

Default value: OFF

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Name of the network profile containing network configurations for the cache redirection virtual server.

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If ACTIVE, respond only if the virtual server is available. If PASSIVE, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

Removes a virtual server.

rm cr vserver <name>@ ...

name

Name of the virtual server to be removed.

rm vserver cr_vip

Changes the specified settings of the cache redirection virtual server.

set cr vserver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-redirect <redirect>] [-onPolicyMatch (CACHE | ORIGIN)] [-precedence (RULE | URL)] [-arp (ON | OFF)] [-via (ON | OFF)] [-dnsVserverName <string>] [-destinationVServer <string>] [-domain <string>] [-reuse (ON | OFF)] [-backupVServer <string>] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-redirectURL <URL>] [-cltTimeout <secs>] [-downStateFlush (ENABLED | DISABLED)] [-l2Conn (ON | OFF)] [-backendssl (ENABLED | DISABLED)] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-

netProfile <string>] [-comment <string>] [-srcIPExpr <expression>] [-originUSIP (ON | OFF)] [-usePortRange (ON | OFF)] [-appflowLog (ENABLED | DISABLED)] [-icmpVsrResponse (PASSIVE | ACTIVE)]

name

Name of the cache redirection virtual server.

IPAddress

New IPv4 or IPv6 address of the cache redirection virtual server. Usually a public IP address. Clients send connection requests to this IP address.

redirect

Type of server to which to redirect HTTP requests. Available settings function as follows: * CACHE - Direct all requests to the cache.* POLICY - Apply the cache redirection policy to determine whether the request should be directed to the cache or to the origin.* ORIGIN - Direct all requests to the origin server.

Possible values: CACHE, POLICY, ORIGIN

Default value: CRD_POLICY

onPolicyMatch

Redirect requests that match the policy to either the cache or the origin server, as specified.

Note: For this option to work, you must set the cache redirection type to POLICY.

Possible values: CACHE, ORIGIN

Default value: CRD_ORIGIN

precedence

Type of policy (URL or RULE) that takes precedence on the cache redirection virtual server. You can use this argument only when configuring cache redirection on the specified virtual server. It applies only if both URL and RULE based policies have been configured on the same virtual server. Available settings function as follows:URL - The incoming request is matched against the URL-based policies before it is matched against the rule-based policies.

For URL based policies, the precedence hierarchy is:

1. Domain and exact URL

2. Domain, prefix and suffix

3. Domain and suffix

4. Domain and prefix

5. Domain only

6. Exact URL

7. Prefix and suffix

8. Suffix only

9. Prefix only

10. Default

RULE - The incoming request is matched against the rule-based policies before it is matched against the URL-based policies.

Possible values: RULE, URL

Default value: CS_PRIORITY_RULE

arp

Use ARP to determine the destination MAC address. Specify OFF to use the incoming destination MAC address, or ON to use ARP to determine the destination MAC address.

Possible values: ON, OFF

via

Insert a via header in each HTTP request. In the case of a cache miss, the request is redirected from the cache server to the origin server. This header indicates whether the request is being sent from a cache server.

Possible values: ON, OFF

Default value: ON

cacheVserver

Name of the default target cache virtual server to which to redirect requests.

dnsVserverName

Name of the DNS virtual server that resolves domain names arriving at the forward proxy virtual server.

Note: This parameter applies only to forward proxy virtual servers, not reverse or transparent.

destinationVServer

Destination virtual server for a transparent or forward proxy cache redirection virtual server.

domain

Default domain for reverse proxies. Domains are configured to direct incoming requests from a specified source domain to a specified target domain. There can be several configured pairs of source and target domains. You can select one pair to be the default. If the host header or URL of an incoming request does not include a source domain, this option sends the request to the specified target domain.

reuse

Reuse TCP connections to the origin server across client connections

Possible values: ON, OFF

Default value: ON

backupVServer

Name of the backup virtual server to which traffic is forwarded if the active server becomes unavailable.

disablePrimaryOnDown

Continue sending traffic to a backup virtual server even after the primary virtual server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectURL

URL of the server to which to redirect traffic if the cache redirection virtual server in the NetScaler becomes unavailable.

cltTimeout

Time-out value, in seconds, after which an idle client connection is terminated.

Maximum value: 31536000

downStateFlush

Perform delayed cleanup of connections to this virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

l2Conn

Use L2 parameters, such as MAC, VLAN, and channel to identify a connection.

Possible values: ON, OFF

backendssl

Decides whether the backend connection made by NS to the origin server will be HTTP or SSL. Applicable only for SSL type CR Forward proxy vserver.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Listenpolicy

String specifying the listen policy for the cache redirection virtual server. Can be either an in-line expression

or the name of a named expression.

Default value: "none"

Listenpriority

Priority of the listen policy specified by the Listen Policy parameter. The lower the number, higher the priority.

Default value: 101

Maximum value: 100

tcpProfileName

Name of the profile containing TCP configuration information for the cache redirection virtual server.

httpProfileName

Name of the profile containing HTTP configuration information for cache redirection virtual server.

netProfile

Name of the network profile containing network configurations for the cache redirection virtual server.

comment

Comments associated with this virtual server.

srcIPExpr

Expression used to extract the source IP addresses from the requests originating from the cache. Can be either an in-line expression or the name of a named expression.

originUSIP

Use the client?s IP address as the source IP address in requests sent to the origin server.

Note: You can enable this parameter to implement fully transparent CR deployment.

Possible values: ON, OFF

Default value: OFF

usePortRange

Use a port number from the port range (set by using the set ns param command, or in the Create Virtual Server (Cache Redirection) dialog box) as the source port in the requests sent to the origin server.

Possible values: ON, OFF

Default value: OFF

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If ACTIVE, respond only if the virtual server is available. If PASSIVE, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

Restores the specified parameters of a cache redirection virtual server to their default values. To unset all except the Name parameter, do not specify a value for any other parameter. Refer to the set cr vserver command for a description of the parameters..Refer to the set cr vserver command for meanings of the arguments.

unset cr vserver <name> [-dnsVserverName] [-destinationVServer] [-domain] [-backupVServer] [-cltTimeout] [-redirectURL] [-l2Conn] [-backendssl] [-originUSIP] [-usePortRange] [-srcIPExpr] [-tcpProfileName] [-httpProfileName] [-appflowLog] [-netProfile] [-icmpVsrResponse] [-redirect] [-onPolicyMatch] [-precedence] [-arp] [-via] [-reuse] [-disablePrimaryOnDown] [-downStateFlush] [-Listenpolicy] [-Listenpriority] [-comment]

Binds a cache redirection policy to a cache redirection virtual server.

bind cr vserver <name> [-lbvserver <string> | (-policyName <string> [-priority <positive_integer>]) | <targetVserver>]

name

Name of the cache redirection virtual server to which to bind the cache redirection policy.

lbvserver

Name of the virtual server to which content is forwarded. Applicable only if the policy is a map policy and the cache redirection virtual server is of type REVERSE.

policyName

Name of the cache redirection policy that you are binding.

Unbinds a cache redirection policy from a cache redirection virtual server.

unbind cr vserver <name> [-policyName <string> | -lbvserver <string>]

name

Name of the cache redirection virtual server from which to unbind the policy.

policyName

Name of the cache redirection policy that you are unbinding.

lbvserver

The virtual server name (created with the add lb vserver command) to which content will be switched.

Default value: "default_lb"

Enables a cache redirection virtual server. Note: Virtual servers, when added, are enabled by default.

enable cr vserver <name>@

name

Name of the cache redirection virtual server to be enabled.

```
enable vserver cr_vip
```

Disables a cache redirection virtual server.

disable cr vserver <name>@

name

Name of the cache redirection virtual server to be disabled. (Because the virtual server is still configured, you can reenable it.)

Note: The appliance still responds to ARP and ping requests sent to the IP address of this virtual server.

```
disable vserver cr_vip
```

Displays cache redirection virtual server information. To display information about all configured cache redirection virtual servers, do not include a parameter. To display detailed information about a specific virtual server, use the name parameter to specify the name of the virtual server.

```
show cr vserver [<name>]
```

name

Name of a cache redirection virtual server about which to display detailed information.

summary

fullValues

format

level

IPAddress

The IP address of the virtual server.

td

Traffic Domain ID

stateflag

value

The ssl card status for the transparent ssl cr vserver.

port

Port number of the virtual server.

range

Number of consecutive IP addresses, starting with the address specified by the IP Address parameter, to include in a range of addresses assigned to this virtual server.

serviceType

Protocol (type of service) handled by the virtual server.

ngname

Nodegroup devno to which this crvserver belongs to

type

Virtual server type.

vsvrcfgflags

Contains the config info of vserver to be used at validation

state

Initial state of the cache redirection virtual server.

status

Status.

cacheType

Mode of operation for the cache redirection virtual server. Available settings function as follows:

* TRANSPARENT - Intercept all traffic flowing to the appliance and apply cache redirection policies to determine whether content should be served from the cache or from the origin server.

* FORWARD - Resolve the hostname of the incoming request, by using a DNS server, and forward requests for non-cacheable content to the resolved origin servers. Cacheable requests are sent to the configured cache servers.

* REVERSE - Configure reverse proxy caches for specific origin servers. Incoming traffic directed to the reverse proxy can either be served from a cache server or be sent to the origin server with or without modification to the URL.

redirect

Type of cache server to which to redirect HTTP requests. Available settings function as follows:

* CACHE - Direct all requests to the cache.

* POLICY - Apply the cache redirection policy to determine whether the request should be directed to the cache or to the origin.

* ORIGIN - Direct all requests to the origin server.

onPolicyMatch

Redirect requests that match the policy to either the cache or the origin server, as specified.

Note: For this option to work, you must set the cache redirection type to POLICY.

precedence

Type of policy (URL or RULE) that takes precedence on the cache redirection virtual server. Applies only to cache redirection virtual servers that have both URL and RULE based policies. If you specify URL, URL based policies are applied first, in the following order:

1. Domain and exact URL

2. Domain, prefix and suffix

3. Domain and suffix

4. Domain and prefix

5. Domain only

6. Exact URL

7. Prefix and suffix

8. Suffix only

9. Prefix only

10. Default

If you specify RULE, the rule based policies are applied before URL based policies are applied.

redirectURL

URL of the server to which to redirect traffic if the cache redirection virtual server configured on the NetScaler appliance becomes unavailable.

authentication

Authentication.

homePage

Home page.

dnsVserverName

Name of the DNS virtual server that resolves domain names arriving at the forward proxy virtual server.

Note: This parameter applies only to forward proxy virtual servers, not reverse or transparent.

domain

Default domain for reverse proxies. Domains are configured to direct an incoming request from a specified source domain to a specified target domain. There can be several configured pairs of source and target domains. You can select one pair to be the default. If the host header or URL of an incoming request does not include a source domain, this option sends the request to the specified target domain.

rule

Rule.

policyName

Policies bound to this vserver.

hits

Number of hits.

serviceName

Service name.

weight

Weight for this service.

cacheVserver

Name of the default cache virtual server to which to redirect requests (the default target of the cache redirection virtual server).NOTE: This attribute is deprecated.The functionality is intact, but we advise to use bind command to do default binding for CR

targetVserver

The CSW target server names.

backupVServer

Name of the backup virtual server to which traffic is forwarded if the

active server becomes unavailable.

priority

The priority for the policy.

cltTimeout

Time-out value, in seconds, after which to terminate an idle client connection.

soMethod

The spillover factor. When the main virtual server reaches this spillover threshold, it will give further traffic to the backupvserver.

soPersistence

The state of spillover persistence.

soPersistenceTimeOut

The spillover persistence entry timeout.

soThreshold

The spillover threshold value.

reuse

Reuse TCP connections to the origin server across client connections. Do not set this parameter unless the Service Type parameter is set to HTTP. If you set this parameter to OFF, the possible settings of the Redirect parameter function as follows:

* CACHE - TCP connections to the cache servers are not reused.

* ORIGIN - TCP connections to the origin servers are not reused.

* POLICY - TCP connections to the origin servers are not reused.

If you set the Reuse parameter to ON, connections to origin servers and connections to cache servers are reused.

arp

destinationVServer

Destination virtual server for a transparent or forward proxy cache redirection virtual server.

via

Insert a via header in each HTTP request. In the case of a cache miss, the request is redirected from the cache server to the origin

server. This header indicates whether the request is being sent from a cache server.

downStateFlush

Perform delayed clean up of connections on this vserver.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

l2Conn

Use L2 parameters, such as MAC, VLAN, and channel to identify a connection.

backendssl

Decides whether the backend connection made by NS to the origin server will be HTTP or SSL. Applicable only for SSL type CR Forward proxy vserver.

comment

Comments associated with this virtual server.

Listenpolicy

The string is listenpolicy configured for CR vserver

Listenpriority

This parameter is the priority for listen policy of CR Vserver.

tcpProfileName

Name of the profile containing TCP configuration information for the cache redirection virtual server.

httpProfileName

Name of the profile containing HTTP configuration information for cache redirection virtual server.

srcIPExpr

Expression used to extract the source IP addresses from the requests originating from the cache. Can be either an in-line expression or the name of a named expression.

originUSIP

Use the client?s IP address as the source IP address in requests

sent to the origin server.

Note: You can enable this parameter to implement fully transparent CR deployment.

usePortRange

Use a port number from the port range (set by using the set ns param command, or in the Create Virtual Server (Cache Redirection) dialog box) as the source port in the requests sent to the origin server.

appflowLog

Enable logging of AppFlow information.

netProfile

Name of the network profile containing network configurations for the cache redirection virtual server.

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If ACTIVE, respond only if the virtual server is available. If PASSIVE, respond even if the virtual server is not available.

lbvserver

The Default target server name.

inherited

On State describes that policy bound is inherited from global binding.

devno

count

Displays statistics for all cache redirection virtual servers or for the cache redirection virtual server specified by the name parameter.

stat cr vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of a specific cache redirection virtual server.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Renames a cache redirection virtual server.

rename cr vserver <name>@ <newName>@

name

Existing name of the cache redirection virtual server.

newName

New name for the cache redirection virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ? my name? or ?my name?).

rename cr vserver vscr1 vscrnew

Content Switching Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- cs action
- cs parameter
- cs policy
- cs policylabel
- cs vserver

cs action

Sep 22, 2015

The following operations can be performed on "cs action":

add | rm | set | unset | show | rename

Creates an action that indicates the target load balancing virtual server. This action is used to specify the target load balancing virtual server while defining a policy to support multiple policy bind support.

add cs action <name> (-targetLBVserver <string> | -targetVserverExpr <expression>) [-comment <string>]

name

Name for the content switching action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the content switching action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my action? or ?my action?).

targetLBVserver

Name of the load balancing virtual server to which the content is switched.

targetVserverExpr

Information about this content switching action.

comment

Comments associated with this cs action.

```
add cs action -targetLBVserver act1 lb1
```

Removes a content switching action.

rm cs action <name>

name

Name of the cs action.

```
rm cs action act_before
```

Modifies the configuration settings of a content switching action.

```
set cs action <name> (-targetLBVserver <string> | -targetVserverExpr <expression>) [-comment <string>]
```

name

Name for the content switching action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the content switching action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my action? or ?my action?).

targetLBVserver

Name of the load balancing virtual server to which the content is switched.

targetVserverExpr

Information about this content switching action.

comment

Comments associated with this cs action.

```
set cs action act1 -targetLBVserver lb2 -comment 'for url'
```

Use this command to remove cs action settings.Refer to the set cs action command for meanings of the arguments.

unset cs action <name> -comment

Displays the configuration settings of the specified content switching action or lists all the content switching actions configured on the appliance.

show cs action [<name>]

name

Name of the content switching action.

summary

fullValues

format

level

stateflag

targetLBVserver

Target LB vserver name.

targetVserverExpr

Target LB vserver expression.

hits

The number of times the action has been taken.

referenceCount

The number of references to the action.

undefHits

The number of times the action resulted in UNDEF.

builtin

comment

Comments associated with this cs action.

devno

count

show cs action

Renames a content switching action.

rename cs action <name>@ <newName>@

name

Existing name of the content switching action.

newName

New name for the content switching action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my name? or ?my name?).

rename cs action oldname newname

---

cs parameter

Sep 22, 2015

The following operations can be performed on "cs parameter":

set | unset | show

Sets the status of the state update parameter for the server. By default, the content switching virtual server is always UP, regardless of the state of the load balancing virtual servers bound to it. This command enables the virtual server to check the status of the attached load balancing server for state information.

set cs parameter -stateupdate (ENABLED | DISABLED)

stateupdate

Specifies whether the virtual server checks the attached load balancing server for state information.

Possible values: ENABLED, DISABLED

Default value: DISABLED

```
set cs parameter -stateupdate (ENABLED|DISABLED)
```

Use this command to remove cs parameter settings.Refer to the set cs parameter command for meanings of the arguments.

```
unset cs parameter -stateupdate
```

Show CS parameters

show cs parameter

format

level

stateupdate

Specifies whether the virtual server checks the attached load balancing server for state information.

show cs parameter

cs policy

Sep 22, 2015

The following operations can be performed on "cs policy":

add | rm | set | unset | show | rename

Creates a new content switching policy. You use this policy to manage content switching on a virtual server.

add cs policy <policyName> [-url <string> | -rule <expression> | -action <string>] [-domain <string>] [-logAction <string>]

policyName

Name for the content switching policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after a policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my policy? or ?my policy?).

url

URL string that is matched with the URL of a request. Can contain a wildcard character. Specify the string value in the following format: [[prefix] [*]] [.suffix].

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

domain

The domain name. The string value can range to 63 characters.

action

Content switching action that names the target load balancing virtual server to which the traffic is switched.

logAction

The log action associated with the content switching policy

To match the requests that have URL "/", you would enter the following command: add cs policy <policyName> -url / To match with all URLs that start with "/sports/", you

---

Removes a content switching policy. You can delete a user-defined content switching policy that is not bound to a content switching virtual server. If the policy is bound to a virtual server, you must first unbind the policy, and then remove it.

rm cs policy <policyName>

policyName

Name of the content switching policy to be removed.

Changes an existing content switching policy.

set cs policy <policyName> [-url <string> | -rule <expression>] [-domain <string>] [-action <string>] [-logAction <string>]

policyName

Name of the content switching policy.

url

The URL, with wildcards.

rule

The condition for applying this policy.

domain

The domain name.

action

The content switching action name.

logAction

The log action associated with the content switching policy

Unset logaction for existing content swtching policy..Refer to the set cs policy command for meanings of the arguments.

unset cs policy <policyName> [-logAction] [-url] [-rule] [-domain] [-action]

unset cs policy pol9 -logAction

Displays all existing content switching policies, or just the specified policy.

show cs policy [<policyName>]

policyName

Name of the content switching policy to display. If this parameter is omitted, details of all the policies are displayed.

summary

fullValues

format

level

url

The URL with wildcards.

rule

The condition for applying this policy.

domain

The domain name.

action

The CS action name.

vstype

Virtual server type.

hits

Total number of hits.

piHits

Total number of hits.

bindHits

Total number of hits.

piPolicyhits

bind hits for PI CS Policy.

labelName

Name of the label invoked.

labelType

The invocation type.

target

Target flagNOTE: This attribute is deprecated.

priority

priority of bound policy

flag

stateflag

activePolicy

Indicates whether policy is bound or not.

csPolicyType

Indicates whether policy is PI or not.(used only during display)

logAction

The log action associated with the content switching policy

devno

count

Rename a content switching policy.

rename cs policy <policyName>@ <newName>@

policyName

The name of the content switching policy.

newName

The new name of the content switching policy.

rename cs policy oldname newname

---

cs policylabel

Sep 22, 2015

The following operations can be performed on "cs policylabel":

add | rm | bind | unbind | show | rename

Adds a content switching policy label.

add cs policylabel <labelName> <cspolicylabeltype>

labelName

Name for the policy label. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

The label name must be unique within the list of policy labels for content switching.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, \\?my label\\? or \\?my policylabel\\?).

cspolicylabeltype

Protocol supported by the policy label. All policies bound to the policy label must either match the specified protocol or be a subtype of that protocol. Available settings function as follows:

* HTTP - Supports policies that process HTTP traffic. Used to access unencrypted Web sites. (The default.)

* SSL - Supports policies that process HTTPS/SSL encrypted traffic. Used to access encrypted Web sites.

* TCP - Supports policies that process any type of TCP traffic, including HTTP.

* SSL_TCP - Supports policies that process SSL-encrypted TCP traffic, including SSL.

* UDP - Supports policies that process any type of UDP-based traffic, including DNS.

* DNS - Supports policies that process DNS traffic.

* ANY - Supports all types of policies except HTTP, SSL, and TCP.

* SIP_UDP - Supports policies that process UDP based Session Initiation Protocol (SIP) traffic. SIP initiates, manages, and terminates multimedia communications sessions, and has emerged as the standard for Internet telephony (VoIP).

* RTSP - Supports policies that process Real Time Streaming Protocol (RTSP) traffic. RTSP provides delivery of multimedia and other streaming data, such as audio, video, and other types of streamed media.

* RADIUS - Supports policies that process Remote Authentication Dial In User Service (RADIUS) traffic. RADIUS supports combined authentication, authorization, and auditing services for network management.

* MYSQL - Supports policies that process MYSQL traffic.

* MSSQL - Supports policies that process Microsoft SQL traffic.

Possible values: HTTP, TCP, RTSP, SSL, SSL_TCP, UDP, DNS, SIP_UDP, ANY, RADIUS, RDP, MYSQL, MSSQL, DIAMETER, SSL_DIAMETER, FTP

```
add cs policylabel trans_http_url HTTP
```

Removes a content switching policy label.

rm cs policylabel <labelName>

labelName

Name of the label to be removed.

```
rm cs policylabel trans_http_url
```

Binds a content switching policy to a content switching policy label.

bind cs policylabel <labelName> <policyName> <priority> [-targetVserver <string> | (-invoke (<labelType> <labelName>))] [-gotoPriorityExpression <expression>]

labelName

Name of the policy label to which to bind a content switching policy.

policyName

Name of the content switching policy to bind to the content switching policy label.

priority

Unsigned integer that determines the priority of the policy relative to other policies in this policy label. Smaller the number, higher the priority.

Minimum value: 1

Maximum value: 2147483647

targetVserver

Name of the virtual server to which to forward requests that match the policy.

gotoPriorityExpression

Expression or other value specifying the priority of the next policy to be evaluated if the current policy rule evaluates to TRUE. Alternatively, you can specify one of the following values:

* NEXT - Go to the policy with the next higher priority.

* END - End evaluation. (This is the default. Evaluation stops if the gotoPriorityExpression parameter is not set.)

* USE_INVOCATION_RESULT - Applicable if this entry invokes another policy label. If the final goto in the invoked policy label has a value of END, evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.

If you specify an expression, its result must be a number. In that case, the next action is determined as follows:

* If the expression evaluates to the priority of a policy with a lower priority (larger priority number) than the current policy, that policy is evaluated next.

* If the expression evaluates to a priority of the current policy, policy with the next highest priority is evaluated.

An UNDEF event is triggered if:

* The expression cannot be evaluated.

* The expression evaluates to a number that is smaller than the highest priority in the policy bank but is not same as any policy's priority.

* The expression evaluates to a number that is smaller than the current policy's priority.

invoke

Invoke other policy labels. After evaluating the policies in the invoked policy label, the appliance continues to evaluate policies that are bound to the current policy label (the selected bind point).

i) bind cs policylabel cs_lab lbvs_1 pol_cs 1 2

Unbinds a content switching policy from a content switching policy label.

unbind cs policylabel <labelName> <policyName>

labelName

Name of the policy label from which to unbind a content switching policy.

policyName

Name of the content switching policy to unbind from the label.

```
unbind cs policylabel cs_lab pol_cs
```

Displays all the content switching policy labels, or just the specified policy label.

```
show cs policylabel [<labelName>]
```

labelName

Name of the content switching policy label to display.

summary

fullValues

format

level

cspolicylabeltype

Protocol supported by the policy label. All policies bound to the policy label must either match the specified protocol or be a subtype of that protocol. Available settings function as follows:

* HTTP - Supports policies that process HTTP traffic. Used to access unencrypted Web sites. (The default.)

* SSL - Supports policies that process HTTPS/SSL encrypted traffic. Used to access encrypted Web sites.

* TCP - Supports policies that process any type of TCP traffic, including HTTP.

* SSL_TCP - Supports policies that process SSL-encrypted TCP traffic, including SSL.

* UDP - Supports policies that process any type of UDP-based traffic, including DNS.

* DNS - Supports policies that process DNS traffic.

* ANY - Supports all types of policies except HTTP, SSL, and TCP.

* SIP_UDP - Supports policies that process UDP based Session Initiation Protocol (SIP) traffic.

SIP initiates, manages, and terminates multimedia communications sessions, and has emerged as the standard for Internet telephony (VoIP).

* RTSP - Supports policies that process Real Time Streaming Protocol (RTSP) traffic. RTSP provides delivery of multimedia and other streaming data, such as audio, video, and other types of streamed media.

* RADIUS - Supports policies that process Remote Authentication Dial In User Service (RADIUS) traffic. RADIUS supports combined authentication, authorization, and auditing services for network management.

* MYSQL - Supports policies that process MYSQL traffic.

* MSSQL - Supports policies that process Microsoft SQL traffic.

stateflag

numpol

number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the content switching policy.

priority

Specifies the priority of the policy.

targetVserver

Name of the virtual server to which to forward requests that match the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

invoke

devno

count

i) show cs policylabel cs_lab ii) show cs policylabel

Rename a content switching policy label.

rename cs policylabel <labelName>@ <newName>@

labelName

The name of the content switching policylabel.

newName

The new name of the content switching policylabel.

rename cs policylabel oldname newname

cs vserver

Sep 22, 2015

The following operations can be performed on "cs vserver":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

Creates a content switching virtual server.

add cs vserver <name> [-td <positive_integer>] <serviceType> ((<IPAddress> [-range <positive_integer>]) | (-IPPattern <ippat> -IPMask <ipmask>)) <port> [-state (ENABLED | DISABLED)] [-stateupdate (ENABLED | DISABLED)] [-cacheable (YES | NO)] [-redirectURL <URL>] [-cltTimeout <secs>] [-precedence (RULE | URL)] [-caseSensitive (ON | OFF)] [-soMethod <soMethod>] [-soPersistence (ENABLED | DISABLED)] [-soPersistenceTimeOut <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-redirectPortRewrite (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-backupVServer <string>] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-insertVserverIPPort <insertVserverIPPort> [<vipHeader>]] [-rtspNat (ON | OFF)] [-AuthenticationHost <string>] [-Authentication (ON | OFF)] [-Listenpolicy <expression> [-Listenpriority <positive_integer>]] [-authn401 (ON | OFF)] [-authnVsName <string>] [-push (ENABLED | DISABLED)] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients (YES | NO)] [-tcpProfileName <string>] [-httpProfileName <string>] [-dbProfileName <string>] [-comment <string>] [-mssqlServerVersion <mssqlServerVersion>] [-l2Conn (ON | OFF)] [-mysqlProtocolVersion <positive_integer>] [-mysqlServerVersion <string>] [-mysqlCharacterSet <positive_integer>] [-mysqlServerCapabilities <positive_integer>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-icmpVsrResponse (PASSIVE | ACTIVE)] [-authnProfile <string>]

name

Name for the content switching virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

Cannot be changed after the CS virtual server is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, \\?my server\\? or \\?my server\\?).

td

Traffic Domain ID

Maximum value: 4094

serviceType

Protocol used by the virtual server.

Possible values: HTTP, SSL, TCP, FTP, RTSP, SSL_TCP, UDP, DNS, SIP_UDP, ANY, RADIUS, RDP, MYSQL, MSSQL, DIAMETER, SSL_DIAMETER

IPAddress

IP address of the content switching virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if the virtual servers, vs1 and vs2, have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

range

Number of consecutive IP addresses, starting with the address specified by the IP Address parameter, to include in a range of addresses assigned to this virtual server.

Default value: 1

Minimum value: 1

Maximum value: 254

port

Port number for content switching virtual server.

Minimum value: 1

state

Initial state of the load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

stateupdate

Enable state updates for a specific content switching virtual server. By default, the Content Switching virtual server is always UP, regardless of the state of the Load Balancing virtual servers bound to it. This parameter interacts with the global setting as follows:

Global Level | Vserver Level | Result

ENABLED ENABLED ENABLED

ENABLED DISABLED ENABLED

DISABLED ENABLED ENABLED

DISABLED DISABLED DISABLED

If you want to enable state updates for only some content switching virtual servers, be sure to disable the state update parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cacheable

Use this option to specify whether a virtual server, used for load balancing or content switching, routes requests to the cache redirection virtual server before sending it to the configured servers.

Possible values: YES, NO

Default value: NO

redirectURL

URL to which traffic is redirected if the virtual server becomes unavailable. The service type of the virtual server should be either HTTP or SSL.

Caution: Make sure that the domain in the URL does not match the domain specified for a content switching policy. If it does, requests are continuously redirected to the unavailable virtual server.

cltTimeout

Idle time, in seconds, after which the client connection is terminated. The default values are:

180 seconds for HTTP/SSL-based services.

9000 seconds for other TCP-based services.

120 seconds for DNS-based services.

120 seconds for other UDP-based services.

Default value: VAL_NOT_SET

Maximum value: 31536000

precedence

Type of precedence to use for both RULE-based and URL-based policies on the content switching virtual server. With the default (RULE) setting, incoming requests are evaluated against the rule-based content switching policies. If none of the rules match, the URL in the request is evaluated against the URL-based content switching policies.

Possible values: RULE, URL

Default value: CS_PRIORITY_RULE

caseSensitive

Consider case in URLs (for policies that use URLs instead of RULES). For example, with the ON setting, the URLs /a/1.html and /A/1.HTML are treated differently and can have different targets (set by content switching policies). With the OFF setting, /a/1.html and /A/1.HTML are switched to the same target.

Possible values: ON, OFF

Default value: ON

soMethod

Type of spillover used to divert traffic to the backup virtual server when the primary virtual server reaches the spillover threshold. Connection spillover is based on the number of connections. Bandwidth spillover is based on the total Kbps of incoming and outgoing traffic.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

Maintain source-IP based persistence on primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Time-out value, in minutes, for spillover persistence.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Depending on the spillover method, the maximum number of connections or the maximum total bandwidth (Kbps) that a virtual server can handle before spillover occurs.

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

State of port rewrite while performing HTTP redirect.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Perform delayed cleanup of connections on this vserver.

Possible values: ENABLED, DISABLED

Default value: ENABLED

backupVServer

Name of the backup virtual server that you are configuring. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the backup virtual server is created. You can assign a different backup virtual server or rename the existing virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks.

disablePrimaryOnDown

Continue forwarding the traffic to backup virtual server even after the primary server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

insertVserverIPPort

Insert the virtual server's VIP address and port number in the request header. Available values function as follows:

VIPADDR - Header contains the vserver's IP address and port number without any translation.

OFF - The virtual IP and port header insertion option is disabled.

V6TOV4MAPPING - Header contains the mapped IPv4 address corresponding to the IPv6 address of the vserver and the port number. An IPv6 address can be mapped to a user-specified IPv4 address using the set ns ip6 command.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

rtspNat

Enable network address translation (NAT) for real-time streaming protocol (RTSP) connections.

Possible values: ON, OFF

Default value: OFF

AuthenticationHost

FQDN of the authentication virtual server. The service type of the virtual server should be either HTTP or SSL.

Authentication

Authenticate users who request a connection to the content switching virtual server.

Possible values: ON, OFF

Default value: OFF

Listenpolicy

String specifying the listen policy for the content switching virtual server. Can be either the name of an existing expression or an in-line expression.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Maximum value: 100

authn401

Enable HTTP 401-response based authentication.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of authentication virtual server that authenticates the incoming user requests to this content switching virtual server.

push

Process traffic with the push virtual server that is bound to this content switching virtual server (specified by the Push VServer parameter). The service type of the push virtual server should be either HTTP or SSL.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the client-facing load balancing virtual server.

pushLabel

Expression for extracting the label from the response received from server. This string can be either an existing rule name or an inline expression. The service type of the virtual server should be either HTTP or SSL.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

tcpProfileName

Name of the TCP profile containing TCP configuration settings for the virtual server.

httpProfileName

Name of the HTTP profile containing HTTP configuration settings for the virtual server. The service type of the virtual server should be either HTTP or SSL.

dbProfileName

Name of the DB profile.

comment

Information about this virtual server.

mssqlServerVersion

The version of the MSSQL server

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: TDS_PROT_2008B

l2Conn

Use L2 Parameters to identify a connection

Possible values: ON, OFF

mysqlProtocolVersion

The protocol version returned by the mysql vserver.

Default value: 10

mysqlServerVersion

The server version string returned by the mysql vserver.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

The character set returned by the mysql vserver.

Default value: 8

mysqlServerCapabilities

The server capabilities returned by the mysql vserver.

Default value: 41613

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

The name of the network profile.

icmpVsrResponse

Can be active or passive

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

authnProfile

Name of the authentication profile to be used when authentication is turned on.

1. You can use precedence when certain client attributes (e.g., browser type) require to be served with different content. All other clients can then be served from conten

---

Removes a content switching virtual server.

rm cs vserver <name>@ ...

name

Name of the virtual server to be removed.

rm vserver cs_vip

Modifies the configuration of a content switching virtual server.

set cs vserver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-IPPattern <ippat>] [-IPMask <ipmask>] [-stateupdate (ENABLED | DISABLED)] [-precedence (RULE | URL)] [-caseSensitive (ON | OFF)] [-backupVServer <string>] [-redirectURL <URL>] [-cacheable (YES | NO)] [-cltTimeout <secs>] [-soMethod <soMethod>] [-soPersistence (ENABLED | DISABLED)] [-soPersistenceTimeOut <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-redirectPortRewrite (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-insertVserverIPPort <insertVserverIPPort> [<vipHeader>]] [-rtspNat (ON | OFF)] [-AuthenticationHost <string>] [-Authentication (ON | OFF)] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>] [-authn401 (ON | OFF)] [-authnVsName <string>] [-push (ENABLED | DISABLED)] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients (YES | NO)] [-tcpProfileName <string>] [-httpProfileName <string>] [-dbProfileName <string>] [-comment <string>] [-l2Conn (ON | OFF)] [-mssqlServerVersion <mssqlServerVersion>] [-mysqlProtocolVersion <positive_integer>] [-mysqlServerVersion <string>] [-mysqlCharacterSet <positive_integer>] [-mysqlServerCapabilities <positive_integer>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-authnProfile <string>] [-icmpVsrResponse (PASSIVE | ACTIVE)]

name

Identifies the virtual server name (created with the add cs vserver command).

IPAddress

The new IP address of the virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if the virtual servers, vs1 and vs2, have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

IPMask

IP mask, in dotted decimal notation, for the IP Pattern parameter. Can have leading or trailing non-zero octets (for example, 255.255.240.0 or 0.0.255.255). Accordingly, the mask specifies whether the first n bits or the last n bits of the destination IP address in a client request are to be matched with the corresponding bits in the IP pattern. The former is called a forward mask. The latter is called a reverse mask.

stateupdate

Enable state updates for a specific content switching virtual server. By default, the Content Switching virtual server is always UP, regardless of the state of the Load Balancing virtual servers bound to it. This parameter interacts with the global setting as follows:

Global Level | Vserver Level | Result

ENABLED ENABLED ENABLED

ENABLED DISABLED ENABLED

DISABLED ENABLED ENABLED

DISABLED DISABLED DISABLED

If you want to enable state updates for only some content switching virtual servers, be sure to disable the state update parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

precedence

The precedence on the content switching virtual server between rule-based and URL-based policies. The default precedence is set to RULE.

If the precedence is configured as RULE, the incoming request is applied against the content switching policies created with the -rule argument. If none of the rules match, then the URL in the request is applied against the content switching policies created with the -url option.

For example, this precedence can be used if certain client attributes (such as a specific type of browser) need to be served different content and all other clients can be served from the content distributed among the servers.

If the precedence is configured as URL, the incoming request URL is applied against the content switching policies created with the -url option. If none of the policies match, then the request is applied against the content switching policies created with the -rule option.

Also, this precedence can be used if some content (such as images) is the same for all clients, but other content (such as text) is different for different clients. In this case, the images will be served to all clients, but the text will be served to specific clients based on specific attributes, such as Accept-Language.

Possible values: RULE, URL

Default value: CS_PRIORITY_RULE

caseSensitive

The URL lookup case option on the content switching vserver.

If case sensitivity of a content switching virtual server is set to 'ON', the URLs /a/1.html and /A/1.HTML are treated differently and may have different targets (set by content switching policies).

If case sensitivity is set to 'OFF', the URLs /a/1.html and /A/1.HTML are treated the same, and will be switched to the same target.

Possible values: ON, OFF

Default value: ON

backupVServer

Name of the backup virtual server that you are configuring. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the backup virtual server is created. You can assign a different backup virtual server or rename the existing virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks.

redirectURL

The redirect URL for content switching.

cacheable

The option to specify whether a virtual server used for content switching will route requests to the cache redirection virtual server before sending it to the configured servers.

Possible values: YES, NO

Default value: NO

cltTimeout

Client timeout in seconds.

Default value: VAL_NOT_SET

Maximum value: 31536000

soMethod

The spillover factor. When traffic on the main virtual server reaches this threshold, additional traffic is sent to the backupvserver.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

Maintain source-IP based persistence on primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

The spillover persistency entry timeout.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Depending on the spillover method, the maximum number of connections or the maximum total bandwidth (Kbps) that a virtual server can handle before spillover occurs.

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

SSL redirect port rewrite.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Perform delayed clean up of connections on this vserver.

Possible values: ENABLED, DISABLED

Default value: ENABLED

disablePrimaryOnDown

Continue forwarding the traffic to backup virtual server even after the primary server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

insertVserverIPPort

The virtual IP and port header insertion option for the vserver.

* VIPADDR - Header contains the vserver's IP address and port number without any translation.

* OFF - The virtual IP and port header insertion option is disabled.

* V6TOV4MAPPING - Header contains the mapped IPv4 address that corresponds to the IPv6 address of the vserver and the port number. An IPv6 address can be mapped to a user-specified IPv4 address using the set ns ip6 command.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

rtspNat

Enable network address translation (NAT) for real-time streaming protocol (RTSP) connections.

Possible values: ON, OFF

Default value: OFF

AuthenticationHost

FQDN of the authentication virtual server. The service type of the virtual server should be either HTTP or SSL.

Authentication

Authenticate users who request a connection to the content switching virtual server.

Possible values: ON, OFF

Default value: OFF

Listenpolicy

String specifying the listen policy for the content switching virtual server. Can be either the name of an existing expression or an in-line expression.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Maximum value: 100

authn401

Enable HTTP 401-response based authentication.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of authentication virtual server that authenticates the incoming user requests to this content switching virtual server.

push

Process traffic with the push virtual server that is bound to this content switching virtual server (specified by the Push VServer parameter). The service type of the push virtual server should be either HTTP or SSL.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the client-facing load balancing virtual server.

pushLabel

Expression for extracting the label from the response received from server. This string can be either an existing rule name or an inline expression. The service type of the virtual server should be either HTTP or SSL.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

tcpProfileName

Name of the TCP profile containing TCP configuration settings for the virtual server.

httpProfileName

Name of the HTTP profile containing HTTP configuration settings for the virtual server. The service type of the virtual server should be either HTTP or SSL.

dbProfileName

Name of the DB profile.

comment

Information about this virtual server.

l2Conn

Use L2 Parameters to identify a connection

Possible values: ON, OFF

mssqlServerVersion

The version of the MSSQL server

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: TDS_PROT_2008B

mysqlProtocolVersion

The protocol version returned by the mysql vserver.

Default value: 10

mysqlServerVersion

The server version string returned by the mysql vserver.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

The character set returned by the mysql vserver.

Default value: 8

mysqlServerCapabilities

The server capabilities returned by the mysql vserver.

Default value: 41613

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

The name of the network profile.

authnProfile

Name of the authentication profile to be used when authentication is turned on.

icmpVsrResponse

Can be active or passive

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

Unset the parameters of a content switching virtual server..Refer to the set cs vserver command for meanings of the arguments.

unset cs vserver <name> [-caseSensitive] [-backupVServer] [-cltTimeout] [-redirectURL] [-authn401] [-Authentication] [-AuthenticationHost] [-authnVsName] [-pushVserver] [-pushLabel] [-tcpProfileName] [-httpProfileName] [-dbProfileName] [-l2Conn] [-mysqlProtocolVersion] [-mysqlServerVersion] [-mysqlCharacterSet] [-mysqlServerCapabilities] [-appflowLog] [-netProfile] [-icmpVsrResponse] [-authnProfile] [-stateupdate] [-precedence] [-cacheable] [-soMethod] [-soPersistence] [-soPersistenceTimeOut] [-soThreshold] [-soBackupAction] [-redirectPortRewrite] [-downStateFlush] [-disablePrimaryOnDown] [-insertVserverIPPort] [-vipHeader] [-rtspNat] [-Listenpolicy] [-Listenpriority] [-push] [-pushMultiClients] [-comment] [-mssqlServerVersion]

Binds a content switching virtual server to a content switching policy.

bind cs vserver <name> [-lbvserver <string> | (-policyName <string> [-targetLBVserver <string>] [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type (REQUEST | RESPONSE)] [-invoke (<labelType> <labelName>)])]

name

Name of the content switching virtual server to which the content switching policy applies.

lbvserver

Name of the default Load Balancing vserver bound. If for a particular content none of the Content Switching policies is evaluated to TRUE, that traffic is switched to default Load Balancing vserver. .

Example: bind cs vserver cs1 -lbvserver lb1

Note: Use this parameter for default binding only.

policyName

Name of the content switching policy to bind to the content switching virtual server Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after a policy is created.

To bind a content switching policy, you need a content-based virtual server (content switching virtual server) and an address-based virtual server (load balancing virtual server). You can assign multiple policies to the virtual server pair.

Note: When binding a CS virtual server to a default LB virtual server, the Policy Name parameter is optional.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my policy? or ?my policy?).

targetVserver

The virtual server name (created with the add lb vserver command) to which content will be switched.

```
i) bind cs vserver csw-vip1 -policyname csw-policy1 -priority 13 ii) bind cs vserver csw-vip2 -policyname csw-ape-policy2 -priority 14 -gotoPriorityExpression NEXT iii)
```

Unbinds the virtual server from the content switching policy.

```
unbind cs vserver <name> [(-policyName <string> [-type ( REQUEST | RESPONSE )]) | -lbvserver <string>] [-priority <positive_integer>]
```

name

Name of the virtual server to unbind from the policy.

policyName

Name of the policy from which to unbind the content switching virtual server. Note: To unbind the content switching virtual server from the default policy, do not specify a value for this parameter.

lbvserver

The virtual server name (created with the add lb vserver command) to which content will be switched.

Default value: "default_lb"

Enables a content switching virtual server.

```
enable cs vserver <name>@
```

name

Name of the content switching virtual server to enable.

Note: Virtual servers, when added, are enabled by default.

```
enable vserver cs_vip
```

Disables a content switching virtual server.

disable cs vserver <name>@

name

Name of the virtual server to be disabled.

disable vserver cs_vip

Displays all existing content switching virtual servers, or just the specified virtual server.

show cs vserver [<name>] show cs vserver stats - alias for 'stat cs vserver'

name

Name of a content switching virtual server for which to display information, including the policies bound to the virtual server. To display a list of all configured Content Switching virtual servers, do not specify a value for this parameter.

summary

fullValues

format

level

insertVserverIPPort

The virtual IP and port header insertion option for the vserver.

vipHeader

The name of virtual IP and port header.

IPAddress

IP address of the content switching virtual server.

td

Traffic Domain ID

IPPattern

The IP address of the virtual server.

IPMask

The IP address mask of the virtual server.

stateflag

value

The ssl card status for the transparent ssl cs vserver.

port

Port number for content switching virtual server.

range

Number of consecutive IP addresses, starting with the address specified by the IP Address parameter, to include in a range of addresses assigned to this virtual server.

serviceType

Protocol used by the virtual server.

ngname

Nodegroup devno to which this csvserver belongs to

type

The bindpoint to which the policy is bound

vsvrcfgflags

Contains the config info of vserver to be used at validation

state

Initial state of the load balancing virtual server.

sc

The state of SureConnect the specified virtual server.

stateupdate

Enable state updates for a specific content switching virtual server. By default, the Content Switching virtual server is always UP, regardless of the state of the Load Balancing virtual servers bound to it. This parameter interacts with the global setting as follows:

Global Level | Vserver Level | Result

ENABLED ENABLED ENABLED

ENABLED DISABLED ENABLED

DISABLED ENABLED ENABLED

DISABLED DISABLED DISABLED

If you want to enable state updates for only some content switching virtual servers, be sure to disable the state update parameter.

status

Status.

cacheType

Cache type.

redirect

Redirect URL string.

precedence

Type of precedence to use for both RULE-based and URL-based policies on the content switching virtual server. With the default (RULE) setting, incoming requests are evaluated against the rule-based content switching policies. If none of the rules match, the URL in the request is evaluated against the URL-based content switching policies.

redirectURL

The redirect URL for content switching.

Authentication

Authentication.

authn401

HTTP 401 response based authentication.

authnVsName

Name of authentication virtual server that authenticates the incoming user requests to this content switching virtual server.

caseSensitive

Consider case in URLs (for policies that use URLs instead of RULES). For example, with the ON setting, the URLs /a/1.html and /A/1.HTML are treated differently and can have different targets (set by content switching policies). With the OFF setting, /a/1.html and /A/1.HTML are switched to the same target.

homePage

Home page.

dnsVserverName

DNS vserver name.

domain

Domain.

rule

Rule.

policyName

Policies bound to this vserver.

hits

Number of hits.

piPolicyhits

Number of hits.

serviceName

Service name.

weight

Weight for this service.

cacheVserver

Cache vserver name.

targetVserver

target vserver name.

backupVServer

Name of the backup virtual server that you are configuring. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the backup virtual server is created. You can assign a different backup virtual server or rename the existing virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks.

priority

Priority for the policy.

cltTimeout

Idle time, in seconds, after which the client connection is terminated. The default values are:

180 seconds for HTTP/SSL-based services.

9000 seconds for other TCP-based services.

120 seconds for DNS-based services.

120 seconds for other UDP-based services.

Listenpolicy

The string is listenpolicy configured for lb vserver

Listenpriority

This parameter is the priority for listen policy of LB Vserver.

soMethod

Type of spillover used to divert traffic to the backup virtual server when the primary virtual server reaches the spillover threshold. Connection spillover is based on the number of connections. Bandwidth spillover is based on the total Kbps of incoming and outgoing traffic.

soPersistence

Maintain source-IP based persistence on primary and backup virtual servers.

soPersistenceTimeOut

Time-out value, in minutes, for spillover persistence.

soThreshold

Depending on the spillover method, the maximum number of connections or the maximum total bandwidth (Kbps) that a virtual server can handle before spillover occurs.

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

cacheable

The state of caching.

url

URL string.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

redirectPortRewrite

Redirect port rewrite.

downStateFlush

Perform delayed clean up of connections on this vserver.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

invoke

Invoke flag.

labelType

The invocation type.

labelName

Name of the label invoked.

gt2GB

This argument has no effect.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimemSec

Time at which last state change happened. Milliseconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

rtspNat

Enable network address translation (NAT) for real-time streaming protocol (RTSP) connections.

AuthenticationHost

FQDN of the authentication virtual server. The service type of the virtual server should be either HTTP or SSL.

push

Process traffic with the push virtual server that is bound to this content switching virtual server (specified by the Push VServer parameter). The service type of the push virtual server should be either HTTP or SSL.

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the client-facing load balancing virtual server.

pushLabel

Expression for extracting the label from the response received from server. This string can be either an existing rule name or an inline expression. The service type of the virtual server should be either HTTP or SSL.

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

tcpProfileName

Name of the TCP profile containing TCP configuration settings for the virtual server.

httpProfileName

Name of the HTTP profile containing HTTP configuration settings for the virtual server. The service type of the virtual server should be either HTTP or SSL.

dbProfileName

Name of the DB profile.

comment

Information about this virtual server.

appfwPolicyFlag

flags

policySubType

oracleServerVersion

Oracle server version

mssqlServerVersion

The version of the MSSQL server

l2Conn

Use L2 Parameters to identify a connection

mysqlProtocolVersion

The protocol version returned by the mysql vserver.

mysqlServerVersion

The server version string returned by the mysql vserver.

mysqlCharacterSet

The character set returned by the mysql vserver.

mysqlServerCapabilities

The server capabilities returned by the mysql vserver.

appflowLog

Enable logging appflow flow information

netProfile

The name of the network profile.

icmpVsrResponse

Can be active or passive

lbvserver

Name of the default lb vserver bound. Use this param for Default binding only. For Example: bind cs vserver cs1 -lbvserver lb1

targetLBVserver

target vserver name.

contentVsvrFlag

authnProfile

Name of the authentication profile to be used when authentication is turned on.

devno

count

Displays statistics of all content switching virtual servers, or statistics for just the specified content switching virtual server.

stat cs vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the content switching virtual server for which to display statistics. To display statistics for all configured Content Switching virtual servers, do not specify a value for this parameter.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Current Client Est connections (ClntEstConn)

Number of client connections in ESTABLISHED state.

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Vserver hits (Hits)

Total vserver hits

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Total Packets rcvd (PktRx)

Total number of packets received by this service or virtual server.

Total Packets sent (PktTx)

Total number of packets sent.

Current client connections (ClntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Spill Over Threshold (SOThresh)

Spill Over Threshold set on the VServer.

Spill Over Hits (NumSo)

Number of times vserver experienced spill over.

Labeled Connection (LblConn)

Number of Labeled connection on this vserver

Push Labeled Connection (PushLbl)

Number of labels for this push vserver.

Deferred Request (DefReq)

Number of deferred request on this vserver

Invalid Request/Response (IvldReqRsp)

Number invalid requests/responses on this vserver

Invalid Request/Response Dropped (IvldReqRspDrp)

Number invalid requests/responses dropped on this vserver

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Renames a content switching virtual server.

rename cs vserver <name>@ <newName>@

name

Existing name of the content switching virtual server.

newName

New name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my name? or ?my name?).

rename cs vserver cs1 cs2

DB Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- db dbProfile
- db user

db dbProfile

Sep 22, 2015

The following operations can be performed on "db dbProfile":

add | rm | set | unset | show

Add a new DB profile on the Netscaler

add db dbProfile <name> [-interpretQuery (YES | NO)] [-stickiness (YES | NO)] [-kcdAccount <string>] [-conMultiplex (ENABLED | DISABLED)] [-enableCachingConMuxOFF (ENABLED | DISABLED)]

name

Name of the DB profile

interpretQuery

Enables/Disables Interpret Query

Possible values: YES, NO

Default value: YES

stickiness

Enables/Disables Stickyness of Query

Possible values: YES, NO

Default value: NO

kcdAccount

Enables/Disables KCD account

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

enableCachingConMuxOFF

Enable Caching With Connection Multiplexing OFF.

Possible values: ENABLED, DISABLED

Default value: DISABLED

add dbprofile <profile name> -interpretQuery YES -stickyness YES -kcdaccount account

---

Remove a DB profile on the Netscaler

rm db dbProfile <name>

name

Name of the DB profile

rm dbprofile <profile name>

Set/modify DB profile values

set db dbProfile <name> [-interpretQuery (YES | NO)] [-stickiness (YES | NO)] [-kcdAccount <string>] [-conMultiplex (ENABLED | DISABLED)] [-enableCachingConMuxOFF (ENABLED | DISABLED)]

name

Name of the DB profile

interpretQuery

Enables/Disables Interpret Query

Possible values: YES, NO

Default value: YES

stickiness

Enables/Disables Stickyness of Query

Possible values: YES, NO

Default value: NO

kcdAccount

Enables/Disables KCD account

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

enableCachingConMuxOFF

Enable Caching With Connection Multiplexing OFF.

Possible values: ENABLED, DISABLED

Default value: DISABLED

```
set dbprofile <profile name> -interpretQuery YES -stickyness YES
```

Unset DB profile values.Refer to the set db dbProfile command for meanings of the arguments.

```
unset db dbProfile <name> [-interpretQuery] [-stickiness] [-kcdAccount] [-conMultiplex] [-enableCachingConMuxOFF]
```

Display all the configured DB profiles in the system. If a name is specified, then only that profile is shown.

```
show db dbProfile [<name>]
```

name

Name of the DB profile.

summary

fullValues

format

level

interpretQuery

Interpret Queries on NS

stickiness

Stickyness for Queries

kcdAccount

KCD account for windows authentication

conMultiplex

KCD account for windows authentication

refCnt

Profile Reference Count

enableCachingConMuxOFF

Enable Caching When Connection Multiplexing is OFF

stateflag

State flag

devno

count

show dbprofile [profile name]

db user

Sep 22, 2015

The following operations can be performed on "db user":

add | rm | set | show

Adds a database user. The user name and password that you specify in this command are added to the nsconfig file and used to authenticate the user.

add db user <userName> {-password }

userName

Name of the database user. Must be the same as the user name specified in the database.

password

Password for logging on to the database. Must be the same as the password specified in the database.

add db user johndoe -password secret

Removes a database user from the NetScaler appliance. Requests from the user are no longer authenticated or routed to the database server.

rm db user <userName>

userName

Name of the database user to remove.

Modifies the password of an existing database user.

set db user <userName>

userName

Name of the database user.

password

The database users password. If you use the CLI, you are prompted for this password after specifying the user name.

set db user johndoe The above command sets the password for johndoe to abcd (Password to be suplied on prompt)

Displays the specified database user or, if no user is specified, all the database users configured on the appliance.

show db user [<userName>] [-loggedIn]

userName

Name of the database user.

loggedIn

Display the names of all database users currently logged on to the NetScaler appliance.

summary

fullValues

format

level

password

Password for logging on to the database. Must be the same as the password specified in the database.

devno

count

stateflag

DNS Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- dns
- dns aaaaRec
- dns action
- dns action64
- dns addRec
- dns cnameRec
- dns global
- dns key
- dns mxRec
- dns nameServer
- dns nsRec
- dns nsecRec
- dns parameter
- dns policy
- dns policy64
- dns policylabel
- dns proxyRecords
- dns ptrRec
- dns records
- dns soaRec
- dns srvRec
- dns stats
- dns suffix
- dns txtRec
- dns view
- dns zone

dns

Sep 22, 2015

The following operations can be performed on "dns":

Displays DNS statistics.

stat dns [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Dns queries (Q)

Total number of DNS queries received.

Multi queries (MtQ)

Total number of Multi Query request received.

Dns responses (Rsp)

Total number of DNS responses received.

Server responses (SvrRsp)

Total number of Server responses received.

Total Record updates (TotRecUp)

Total number of record updates.

Auth answers (AuthAns)

Number of queries which were authoritatively answered.

Server queries (SvrQ)

Total number of Server queries sent.

Cache flush called (CaFsh)

Total number of times cache was flushed.

Cache entries flushed (CaEntFsh)

Total number of cache entries flushed.

Non-authoritative entries (PxyEnt)

Total number of non-authoritative entries.

Authoritative entries (AthEnt)

Total number of authoritative entries.

Nonexistent domain (NoDomain)

Number of queries for which no record was found.

Response class unsupported (RspClsEr)

Total number of responses for which response types were unsupported.

Invalid query format (InQFmt)

Total number of queries whose format was invalid.

Stray answers (StryRsp)

Total number of stray answers.

Incorrect RD length (BadRDlen)

Number of DNS responses received with invalid resoure data length.

Requests refused (ReqRefused)

Number of DNS requests refused.

NULL Attack (NullAttack)

Total number of queries received where all the counts are 0.

Response type unsupported (RspNoSup)

Total number of responses for which response type requested was unsupported.

Query class unsupported (QClsEr)

Total number of queries for which query class was unsupported.

Invalid response format (InRspFmt)

Total number of responses for which there was a format error.

No answer responses (NoAnswer)

Number of DNS responses received without answer.

Multi queries disabled (MtQErr)

Total number of times a multi query was disabled and received a multi query.

Other errors (OtherErr)

Total number of other errors.

DNS64 queries

Total number of DNS64 queries recieved.

DNS64 answers

Total number of DNS64 answers served.

DNS64 rewrite answers

Total number of DNS64 answers served after rewriting the response.

DNS64 responses

Total number of responses recieved from backend in DNS64 context.

DNS64 GSLB Queries

Total number of DNS64 queries for GSLB domain

DNS64 GSLB Answers

Total number of DNS64 queries served.

DNS64 Total truncated answers

Total number of Answers served with TC bit set in DNS64 context.

DNS64 Total A queries to server

Total number of Queries sent by DNS64 module to backend.

DNS64 Total times AAAA query bypassed

Total number of times AAAA query has been bypassed in DNS64 trnsaction.

DNS64 Total TCP queries

Total number of dns64 queries over TCP

DNS64 Total Active policies

Total number of active dns64 policies

DNS64 Total NODATA Responses

Total number of responses recieved from backend with ancount 0

NS queries (NSQ)

Total number of NS queries received.

SOA queries (SOAQ)

Total number of SOA queries received.

PTR queries (PTRQ)

Total number of PTR queries received.

SRV queries (SRVQ)

Total number of SRV queries received.

A responses (ARsp)

Total number of A responses received.

CNAME responses (CNRsp)

Total number of CNAME responses received.

MX responses (MXRsp)

Total number of MX responses received.

ANY responses (ANYRsp)

Total number of ANY responses received.

NS updates (NSUp)

Total number of NS record updates.

SOA updates (SOAUp)

Total number of SOA record updates.

PTR updates (PTRUp)

Total number of PTR record updates.

SRV updates (SRVUp)

Total number of SRV record updates.

AAAA queries (AAAAQ)

Total number of AAAA queries received.

A queries (AQ)

Total number of A queries received.

CNAME queries (CNQ)

Total number of CNAME queries received.

MX queries (MXQ)

Total number of MX queries received.

ANY queries (ANYQ)

Total number of ANY queries received.

AAAA responses (AAAARsp)

Total number of AAAA responses received.

NS responses (NSRsp)

Total number of NS responses received.

SOA responses (SOARsp)

Total number of SOA responses received.

PTR responses (PTRRsp)

Total number of PTR responses received.

SRV responses (SRVRsp)

Total number of SRV responses received.

AAAA updates (AAAAUp)

Total number of AAAA record updates.

A updates (AUp)

Total number of A record updates.

MX updates (MXUp)

Total number of MX record updates.

CNAME updates (CNUp)

Total number of CNAME record updates.

AAAA records (AAAARec)

Total number of AAAA records.

A records (ARec)

Total number of A records.

MX records (MXRec)

Total number of MX records.

CNAME records (CNRec)

Total number of CNAME records.

NS records (NSRec)

Total number of NS records.

SOA records (SOARec)

Total number of SOA records.

PTR records (PTRRec)

Total number of PTR records.

SRV records (SRVRec)

Total number of SRV records.

No AAAA records (NoAAAARec)

Total number of times AAAA record lookup failed.

No A records (NoARec)

Total number of times A record lookup failed.

No MX records (NoMXRec)

Total number of times MX record lookup failed.

No PTR records (NoPTRRec)

Total number of times PTR record lookup failed.

No NS records (NoNSRec)

Total number of times NS record lookup failed.

No CNAME records (NoCNRec)

Total number of times CNAME record lookup failed.

No SOA records (NoSOARec)

Total number of times SOA record lookup failed.

No SRV records (NoSRVRec)

Total number of times SRV record lookup failed.

No ANY records (NoANYrec)

Total number of times ANY query lookup failed.

Unsupported queries (NotSupQ)

Total number of requests for which query type requested was unsupported.

dns aaaaRec

Sep 22, 2015

The following operations can be performed on "dns aaaaRec":

add | rm | show

Creates a AAAA address record for the specified domain name. You cannot modify a AAAA address record.

add dns aaaaRec <hostName> <IPv6Address> ... [-TTL <secs>]

hostName

Domain name.

IPv6Address

One or more IPv6 addresses to assign to the domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

add dns aaaarec www.mynw.com 3::4:5 -ttl 10

Removes an IPv6 address from a AAAA address record. The associated domain name must be specified. If no IPv6 address is specified, all AAAA records that belong to the specified domain name are removed.

rm dns aaaaRec <hostName> [<IPv6Address> ...]

hostName

Domain name.

IPv6Address

IPv6 address(es) of the AAAA record(s) to remove from the specified domain name.

rm dns aaaarec www.mynw.com

Displays the AAAA (IPv6) address record for the specified host name. If a hostname is not specified, all configured AAAA records are shown.

show dns aaaaRec [<hostName> | -type <type>] [<IPv6Address>]

hostName

Domain name.

IPv6Address

One or more IPv6 addresses to assign to the domain name.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

vServerName

Vitual server name.

authType

Authentication type.

devno

count

stateflag

dns action

Sep 22, 2015

The following operations can be performed on "dns action":

add | rm | set | unset | show

Add a dns action.

add dns action <actionName> <actionType> [-IPAddress <ip_addr|ipv6_addr> ... | -viewName <string> | -preferredLocList <string> ...] [-TTL <secs>]

actionName

Name of the dns action.

actionType

The type of DNS action that is being configured.

Possible values: ViewName, GslbPrefLoc, Drop, Cache_Bypass, Rewrite_Response

IPAddress

List of IP address to be returned in case of rewrite_response actiontype. They can be of IPV4 or IPV6 type.

In case of set command We will remove all the IP address previously present in the action and will add new once given in set dns action command.

TTL

Time to live, in seconds.

Default value: 3600

Maximum value: 2147483647

viewName

The view name that must be used for the given action.

preferredLocList

The location list in priority order used for the given action.

add dns action <actionName> <actionType> (-IPAddress <ip_addr|ipv6_addr> ... | -viewName <string> | -preferredLocList <string> ...) [-TTL <secs>] add dns action a

---

Removes a dns Action.

rm dns action <actionName>

actionName

Name of the dns action.

rm dns action action1

Set a dns Action. Use this command to set the values for Ip address and TTL, If Ipaddress is given in set dns action command we will discard the previous set and will apply this new set of ipaddress given.

set dns action <actionName> [-IPAddress <ip_addr|ipv6_addr> ...] [-TTL <secs>] [-viewName <string>] [-preferredLocList <string> ...]

actionName

Name of the dns action.

IPAddress

List of IP address to be returned in case of rewrite_response actiontype. They can be of IPV4 or IPV6 type.

In case of set command We will remove all the IP address previously present in the action and will add new once given in set dns action command.

TTL

Time to live, in seconds.

Default value: 3600

Maximum value: 2147483647

viewName

The view name that must be used for the given action.

preferredLocList

The location list in priority order used for the given action.

set dns action <actionName> [-IPAddress <ip_addr|ipv6_addr> ...] [-TTL <secs>] [-viewName <string>] [-preferredLocList <string> ...] set dns action action1 -ipAddre

---

Use this command to remove dns action settings.Refer to the set dns action command for meanings of the arguments.

unset dns action <actionName> -TTL

Used to display the action-related information.

show dns action [<actionName>]

actionName

Name of the dns action.

format

level

actionType

The type of DNS action that is being configured.

TTL

Time to live, in seconds.

IPAddress

List of IP address to be returned in case of rewrite_response actiontype. They can be of IPV4 or IPV6 type.

In case of set command We will remove all the IP address previously present in the action and will add new once given in set dns action command.

viewName

The view name that must be used for the given action.

preferredLocList

The location list in priority order used for the given action.

drop

The dns packet must be dropped.

cacheBypass

By pass dns cache for this.

builtin

Flag to determine whether DNS action is default or not

devno

count

stateflag

show dns action <Action-Name> show dns action action1 show dns action

dns action64

Sep 22, 2015

The following operations can be performed on "dns action64":

add | rm | set | unset | show

Add a dns64 action.

add dns action64 <actionName> -Prefix <ipv6_addr|*> [-mappedRule <expression>] [-excludeRule <expression>]

actionName

Name of the dns64 action.

Prefix

The dns64 prefix to be used if the after evaluating the rules

mappedRule

The expression to select the criteria for ipv4 addresses to be used for synthesis.

Only if the mappedrule is evaluated to true the corresponding ipv4 address is used for synthesis using respective prefix,

otherwise the A RR is discarded

excludeRule

The expression to select the criteria for eliminating the corresponding ipv6 addresses from the response.

add dns dns64action <actionName> -prefix f23d:f43e::0/32 [-mappedRule <expr>] [-excludeRule <expr>]

Removes a dns64 Action.

rm dns action64 <actionName>

actionName

Name of the dns64 action.

rm dns dns64action action1

Set a DNS64 Action

set dns action64 <actionName> [-Prefix <ipv6_addr|*>] [-mappedRule <expression>] [-excludeRule <expression>]

actionName

Name of the dns64 action.

Prefix

The dns64 prefix to be used if the after evaluating the rules

mappedRule

The expression to select the criteria for ipv4 addresses to be used for synthesis.

Only if the mappedrule is evaluated to true the corresponding ipv4 address is used for synthesis using respective prefix,

otherwise the A RR is discarded

excludeRule

The expression to select the criteria for eliminating the corresponding ipv6 addresses from the response.

set dns dns64action -prefix -mappedrule -excluderule

Use this command to remove dns action64 settings.Refer to the set dns action64 command for meanings of the arguments.

unset dns action64 <actionName> [-Prefix] [-mappedRule] [-excludeRule]

Used to display the action-related information.

show dns action64 [<actionName>]

actionName

Name of the dns64 action.

format

level

Prefix

The dns64 prefix to be used if the after evaluating the rules

mappedRule

The expression to select the criteria for ipv4 addresses to be used for synthesis.

Only if the mappedrule is evaluated to true the corresponding ipv4 address is used for synthesis using respective prefix,

otherwise the A RR is discarded

excludeRule

The expression to select the criteria for eliminating the corresponding ipv6 addresses from the response.

builtin

Flag to determine whether dna64action is default or not

devno

count

stateflag

show dns dns64action

dns addRec

Sep 22, 2015

The following operations can be performed on "dns addRec":

add | rm | show

Creates an IPv4 address record for the specified domain name. You cannot modify an address resource record.

add dns addRec <hostName> <IPAddress> ... [-TTL <secs>]

hostName

Domain name.

IPAddress

One or more IPv4 addresses to assign to the domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Add dns addrec www.mynw.com 65.200.211.139 -ttl 10

Removes an IPv4 address from an address record. The associated domain name must be specified. If no IPv4 address is specified, all records that belong to the specified domain name are removed.

rm dns addRec <hostName> [<IPAddress> ...]

hostName

Domain name.

IPAddress

IPv4 address(es) of the address records to remove from the specified domain name.

rm dns addrec www.mynw.com

Displays the IPv4 address record for the specified host name. If a hostname is not specified, all configured address records are shown.

show dns addRec [<hostName> | -type <type>]

hostName

Domain name.

type

The address record type. The type can take 3 values:

ADNS - If this is specified, all of the authoritative address records will be displayed.

PROXY - If this is specified, all of the proxy address records will be displayed.

ALL - If this is specified, all of the address records will be displayed.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

IPAddress

IP addresses for the domain name.

TTL

The time to live, in seconds.

vServerName

Vitual server name.

authType

Authentication type.

devno

count

stateflag

dns cnameRec

Sep 22, 2015

The following operations can be performed on "dns cnameRec":

add | rm | show

Creates a canonical name (CNAME) record, or alias, for the specified domain name.

add dns cnameRec <aliasName> <canonicalName> [-TTL <secs>]

aliasName

Alias for the canonical domain name.

canonicalName

Canonical domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

add dns cnameRec www.mynw.org www.mynw.com -ttl 20

Removes a canonical name (CNAME) record.

rm dns cnameRec <aliasName>

aliasName

Alias for which to remove the CNAME record.

rm dns cnamerec www.mynw.org

Displays the canonical name (CNAME) records configured for the specified alias. If no alias is specified, all configured CNAME records are displayed

show dns cnameRec [<aliasName> | -type <type>]

aliasName

Alias for which to display CNAME records.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: NSDNS_AUTH_HOST

summary

fullValues

format

level

canonicalName

Canonical domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be

cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

vServerName

GSLB Vitual server name to which this domain is bound

authType

Record type.

devno

count

stateflag

show dns cnameRec www.mynw.org

dns global

Sep 22, 2015

The following operations can be performed on "dns global":

bind | unbind | show

Binds the specified DNS policy globally.

bind dns global <policyName> <priority> [-gotoPriorityExpression <string>] [-type <type>] [-invoke (<labelType> <labelName>)]

policyName

Name of the DNS policy to bind globally.

bind dns global pol9 9

Unbinds the specified DNS policy from the global bind point.

unbind dns global <policyName>

policyName

Name of the DNS policy to unbind.

unbind dns global pol9

Displays the DNS policies bound to the specified global bind point. If a global bind point is not specified, the command displays the global bind points that have policies bound to them, and the number of policies bound to each of those bind points.

show dns global [-type <type>]

type

Type of global bind point for which to show bound policies.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

summary

fullValues

format

level

stateflag

policyName

Name of the dns policy.

priority

Specifies the priority of the policy with which it is bound. Maximum allowed priority should be less than 65535

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

* NEXT - Evaluate the policy with the next higher priority number.

* END - End policy evaluation.

* USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.

* A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

* If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.

* If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.

* If the expression evaluates to a priority number that is numerically higher than the highest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

* The expression is invalid.

* The expression evaluates to a priority number that is numerically lower than the current policy's priority.

* The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

Invoke flag.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

numpol

The number of policies bound to the bindpoint.

flowType

flowtype of the bound rewrite policy.

flags

upgraded

It is internally used to tell that the policy is a upgraded policy.

builtin

Flag to determine whether DNS policy binding is default or not

devno

count

show dns global show dns global -type REQ_DEFAULT show dns global -type RES_DEFAULT

dns key

Sep 22, 2015

The following operations can be performed on "dns key":

add | create | set | unset | rm | show

Adds a DNS key to the zone that is specified in the key file.

add dns key <keyName> <publickey> <privatekey> [-expires <positive_integer> [<units>]] [-notificationPeriod <positive_integer> [<units>]] [-TTL <secs>]

keyName

Name of the public-private key pair to publish in the zone.

publickey

File name of the public key.

privatekey

File name of the private key.

expires

Time period for which to consider the key valid, after the key is used to sign a zone.

Default value: 120

Minimum value: 1

Maximum value: 32767

notificationPeriod

Time at which to generate notification of key expiration, specified as number of days, hours, or minutes before expiry. Must be less than the expiry period. The notification is an SNMP trap sent to an SNMP manager. To enable the appliance to send the trap, enable the DNSKEY-EXPIRY SNMP alarm.

Default value: 7

Minimum value: 1

Maximum value: 32767

TTL

Time to Live (TTL), in seconds, for the DNSKEY resource record created in the zone. TTL is the time for which the record must be cached by the DNS proxies. If the TTL is not specified, either the DNS zone's minimum TTL or the default value of 3600 is used.

Default value: 3600

Maximum value: 2147483647

add dns key secure.example.zsk -public secure.example-rsasha1-1024.key -private /nsconfig/dns/secure.example-rsasha1-1024.private

---

Creates a public-private key pair to use for signing a DNS zone. The keys are created in the /nsconfig/dns/ directory on the NetScaler appliance. The private, pubic, and DS key files are created with names having the format <prefix>.<key/private/ds>.

create dns key -zoneName <string> -keyType <keyType> -algorithm RSASHA1 -keySize <positive_integer> -fileNamePrefix <string>

zoneName

Name of the zone for which to create a key.

keyType

Type of key to create.

Possible values: KSK, KeySigningKey, ZSK, ZoneSigningKey

Default value: NS_DNSKEY_ZSK

algorithm

Algorithm to generate for zone signing.

Possible values: RSASHA1

Default value: NS_DNSKEYALGO_RSASHA1

keySize

Size of the key, in bits.

Default value: 512

fileNamePrefix

Common prefix for the names of the generated public and private key files and the Delegation Signer (DS) resource record. During key generation, the .key, .private, and .ds suffixes are appended automatically to the file name prefix to produce the names of the public key, the private key, and the DS record, respectively.

create dns key -zone dnssec.bar -algorithm RSASHA1 -keySize 1024

Modifies the specified parameters of a DNS key. Note: If you change the expiry time period of a key, the NetScaler appliance, using the modified key, automatically re-signs all the resource records in the zone, provided that the zone is currently signed with the particular key.

set dns key <keyName> [-expires <positive_integer> [<units>]] [-notificationPeriod <positive_integer> [<units>]] [-TTL <secs>]

keyName

Name of the public-private key pair.

expires

Time period for which to consider the key valid, after the key is used to sign a zone.

Default value: 120

Minimum value: 1

Maximum value: 32767

notificationPeriod

Time at which to generate notification of key expiration, specified as number of days, hours, or minutes before expiry. Must be less than the expiry

period. The notification is an SNMP trap sent to an SNMP manager. To enable the appliance to send the trap, enable the DNSKEY-EXPIRY SNMP alarm.

Default value: 7

Minimum value: 1

Maximum value: 32767

TTL

Time to Live (TTL), in seconds, for the DNSKEY resource record created in the zone. TTL is the time for which the record must be cached by the DNS proxies. If the TTL is not specified, either the DNS zone's minimum TTL or the default value of 3600 is used.

Default value: 3600

Maximum value: 2147483647

```
add dns key secure.example.zsk  -public secure.example-rsasha1-1024.key -private /nsconfig/dns/secure.example-rsasha1-1024.private
```

Use this command to remove dns key settings.Refer to the set dns key command for meanings of the arguments.

```
unset dns key <keyName> [-expires] [-units] [-notificationPeriod] [-units] [-TTL]
```

Removes a DNS key.

```
rm dns key <keyName>
```

keyName

Name of the public-private key pair.

```
rm dns key secure.example.zsk
```

Displays the parameters of the specified DNS key. If no DNS key name is specified, all configured DNS keys are shown. Note: You cannot view the parameters of a public/private key file. You can view the parameters of a key after you have published it in a DNS zone by using either the add dns key command or the DNS > Zones > Sign/Unsign DNS Zone dialog box.

```
show dns key [<keyName>]
```

keyName

Name of the public-private key pair.

summary

fullValues

format

level

publickey

File name of the public key.

privatekey

File name of the private key.

expires

Number of days since signing with this key, when the key expires.

units

Units for the notification period.

notificationPeriod

Time at which to generate notification of key expiration, specified as number of days, hours, or minutes before expiry. Must be less than the expiry period. The notification is an SNMP trap sent to an SNMP manager. To enable the appliance to send the trap, enable the DNSKEY-EXPIRY SNMP alarm.

TTL

Time to Live (TTL), in seconds, for the DNSKEY resource record created in the zone. TTL is the time for which the record must be cached by the DNS proxies. If the TTL is not specified, either the DNS zone's minimum TTL or the default value of 3600 is used.

devno

count

stateflag

show dns key

dns mxRec

Sep 22, 2015

The following operations can be performed on "dns mxRec":

add | rm | set | unset | show

Creates a mail exchange (MX) record for the specified domain name.

add dns mxRec <domain> -mx <string> -pref <positive_integer> [-TTL <secs>]

domain

Domain name for which to add the MX record.

mx

Host name of the mail exchange server.

pref

Priority number to assign to the mail exchange server. A domain name can have multiple mail servers, with a priority number assigned to each server. The lower the priority number, the higher the mail server's priority. When other mail servers have to deliver mail to the specified domain, they begin with the mail server with the lowest priority number, and use other configured mail servers, in priority order, as backups.

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Removes the specified mail exchange (MX) record from the specified domain.

rm dns mxRec <domain> <mx>

domain

Domain name.

mx

Host name of the mail exchange server.

Modifies the priority number and TTL of the mail exchange (MX) record.

set dns mxRec <domain> -mx <string> [-pref <positive_integer>] [-TTL <secs>]

domain

Domain of the MX record to be modified.

mx

Host name of the mail exchange server to be modified.

pref

Priority number to assign to the mail exchange server. A domain name can have multiple mail servers, with a priority number assigned to each server. The lower the priority number, the higher the mail server's priority. When other mail servers have to deliver mail to the specified domain, they begin with the mail server with the lowest priority number, and use other configured mail servers, in priority order, as backups.

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Use this command to remove dns mxRec settings.Refer to the set dns mxRec command for meanings of the arguments.

unset dns mxRec <domain> -mx <string> -TTL

Displays the mail exchange (MX) records for the specified domain. If no domain name is specified, all configured mail exchange records are shown.

show dns mxRec [<domain> | -type <type>]

domain

Domain name.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: NSDNS_AUTH_HOST

summary

fullValues

format

level

mx

Host name of the mail exchange server.

pref

Priority number to assign to the mail exchange server. A domain name can have multiple mail servers, with a priority number assigned to each server. The lower the priority number, the higher the mail server's priority. When other mail servers have to deliver mail to the specified domain, they begin with the mail server with the lowest priority number, and use other configured mail servers, in priority order, as backups.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns nameServer

Sep 22, 2015

The following operations can be performed on "dns nameServer":

add | rm | enable | disable | show

Adds a name server to the appliance. Following are the two types of name servers that can be added: * IP address-based name server - An external name server to contact for domain name resolution. If multiple IP address-based name servers are configured on the appliance, and the local parameter is not set on any of them, incoming DNS queries are load balanced across all the name servers, in round robin fashion. * Virtual server-based name server - A DNS virtual server configured in the NetScaler appliance. If you want more fine-grained control on how external DNS name servers are load balanced (for example, you want a load balancing method other than round robin), you configure a DNS virtual server on the appliance, bind the external name servers as its services, and then specify the name of the virtual server in this command.

add dns nameServer ((<IP> [-local]) | <dnsVserverName>) [-state (ENABLED | DISABLED)] [-type <type>]

IP

IP address of an external name server or, if the Local parameter is set, IP address of a local DNS server (LDNS).

dnsVserverName

Name of a DNS virtual server. Overrides any IP address-based name servers configured on the NetScaler appliance.

local

Mark the IP address as one that belongs to a local recursive DNS server on the NetScaler appliance. The appliance recursively resolves queries received on an IP address that is marked as being local. For recursive resolution to work, the global DNS parameter, Recursion, must also be set.

If no name server is marked as being local, the appliance functions as a stub resolver and load balances the name servers.

state

Administrative state of the name server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

type

Protocol used by the name server. UDP_TCP is not valid if the name server is a DNS virtual server configured on the appliance.

Possible values: UDP, TCP, UDP_TCP

Default value: NSA_UDP

Adding an-IP based nameserver IP: add nameserver 10.102.4.1, Adding a vserver-based name server: add nameserver dns_vsvr where dns_vsvr is the name of a DNS vser

---

Removes a name server from the NetScaler appliance. If the name server is an IP-address based external name server, the name server entry is removed. If the name server is a DNS virtual server on the appliance, the virtual server is not removed, but it is no longer used to resolve domain names.

rm dns nameServer (<IP> | <dnsVserverName>)

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

Deleting an IP-based nameserver: rm nameserver 10.102.4.1, Deleting a vserver-based nameserver: rm nameserver dns_vsvr

---

Enables a name server.

enable dns nameServer (<IP> | <dnsVserverName>)

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

```
enable dns nameserver 10.14.43.149
```

Disables a name server.

disable dns nameServer (<IP> | <dnsVserverName>)

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

```
disable dns nameserver 10.14.43.149
```

Displays the name servers configured on the NetScaler appliance, along with their administrative states.

show dns nameServer [<IP> | <dnsVserverName>]

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

summary

fullValues

format

level

serviceName

The name of the dns vserver.

port

Port of the service.

type

Protocol used by the name server. UDP_TCP is not valid if the name server is a DNS virtual server configured on the appliance.

state

Administrative state of the name server.

nameserverstate

State of the server.

local

ip is a local recursive nameserver.

adminState

ClMonOwner

Tells the mon owner of the service.

ClMonView

Tells the view id by which state of the service is updated.

devno

count

stateflag

dns nsRec

Sep 22, 2015

The following operations can be performed on "dns nsRec":

add | rm | show

Creates a name server record for the specified domain.

add dns nsRec <domain> <nameServer> [-TTL <secs>]

domain

Domain name.

nameServer

Host name of the name server to add to the domain.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Removes the specified name server record from the specified domain.

rm dns nsRec <domain> <nameServer>

domain

Domain name.

nameServer

Name server to remove.

Displays the name server records for the specified domain. If no domain name is specified, all configured name server records are shown.

show dns nsRec [<domain> | -type <type>]

domain

Domain name.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

nameServer

Host name of the name server to add to the domain.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not

available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns nsecRec

Sep 22, 2015

The following operations can be performed on "dns nsecRec":

Displays the NextSECure (NSEC) resource records created for the specified domain name.

show dns nsecRec [<hostName> | -type <type>]

hostName

Name of the domain.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

summary

fullValues

nextNsec

Next nsec record in the chain

nsecBitarray

Bit array representing the different record types configured for the domain nameNOTE: This attribute is deprecated.This is deprecated attribute.

nextRecs

An array of record types associated with the nsec record.

TTL

Time to Live (TTL), in seconds, for the record.

devno

count

stateflag

show dns nsecRec foo.bar

---

dns parameter

Sep 22, 2015

The following operations can be performed on "dns parameter":

set | unset | show

Modifies global DNS parameters on the NetScaler appliance.

set dns parameter [-retries <positive_integer>] [-minTTL <secs>] [-maxTTL <secs>] [-cacheRecords (YES | NO)] [-nameLookupPriority (WINS | DNS)] [-recursion (ENABLED | DISABLED)] [-resolutionOrder <resolutionOrder>] [-dnssec (ENABLED | DISABLED)] [-maxPipeline <positive_integer>] [-dnsRootReferral (ENABLED | DISABLED)] [-dns64Timeout <msecs>]

retries

Maximum number of retry attempts when no response is received for a query sent to a name server. Applies to end resolver and forwarder configurations.

Default value: 5

Minimum value: 1

Maximum value: 5

minTTL

Minimum permissible time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is lower than the value configured for minTTL, the TTL of the record is set to the value of minTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

Maximum value: 604800

maxTTL

Maximum time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is higher than the value configured for maxTTL, the TTL of the record is set to the value of maxTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

Default value: 604800

Minimum value: 1

Maximum value: 604800

cacheRecords

Cache resource records in the DNS cache. Applies to resource records obtained through proxy configurations only. End resolver and forwarder configurations always cache records in the DNS cache, and you cannot disable this behavior. When you disable record caching, the appliance stops caching server responses. However, cached records are not flushed. The appliance does not serve requests from the cache until record caching is enabled again.

Possible values: YES, NO

Default value: YES

nameLookupPriority

Type of lookup (DNS or WINS) to attempt first. If the first-priority lookup fails, the second-priority lookup is attempted. Used only by the SSL VPN feature.

Possible values: WINS, DNS

Default value: NS_WINSFIRST

recursion

Function as an end resolver and recursively resolve queries for domains that are not hosted on the NetScaler appliance. Also resolve queries recursively when the external name servers configured on the appliance (for a forwarder configuration) are unavailable. When external name servers are unavailable, the appliance queries a root server and resolves the request recursively, as it does for an end resolver configuration.

Possible values: ENABLED, DISABLED

Default value: DISABLED

resolutionOrder

Type of DNS queries (A, AAAA, or both) to generate during the routine functioning of certain NetScaler features, such as SSL VPN, cache redirection, and the integrated cache. The queries are sent to the external name servers that are configured for the forwarder function. If you specify both query types, you can also specify the order. Available settings function as follows:

* OnlyAQuery. Send queries for IPv4 address records (A records) only.

* OnlyAAAAQuery. Send queries for IPv6 address records (AAAA records) instead of queries for IPv4 address records (A records).

* AThenAAAAQuery. Send a query for an A record, and then send a query for an AAAA record if the query for the A record results in a NODATA response from the name server.

* AAAAThenAQuery. Send a query for an AAAA record, and then send a query for an A record if the query for the AAAA record results in a NODATA response from the name server.

Possible values: OnlyAQuery, OnlyAAAAQuery, AThenAAAAQuery, AAAAThenAQuery

Default value: NS_FOUR

dnssec

Enable or disable the Domain Name System Security Extensions (DNSSEC) feature on the appliance. Note: Even when the DNSSEC feature is enabled, forwarder configurations (used by internal NetScaler features such as SSL VPN and Cache Redirection for name resolution) do not support the DNSSEC OK (DO) bit in the EDNS0 OPT header.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxPipeline

Maximum number of concurrent DNS requests to allow on a single client connection, which is identified by the <clientip:port>-<vserver ip:port> tuple. A value of 0 (zero) applies no limit to the number of concurrent DNS requests allowed on a single client connection.

Default value: NSNATPCB_MAXPIPELINE

dnsRootReferral

Send a root referral if a client queries a domain name that is unrelated to the domains configured/cached on the NetScaler appliance. If the setting is disabled, the appliance sends a blank response instead of a root referral. Applicable to domains for which the appliance is authoritative. Disable the parameter when the appliance is under attack from a client that is sending a flood of queries for unrelated domains.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dns64Timeout

While doing DNS64 resolution, this parameter specifies the time to wait before sending an A query if no response is received from backend DNS server for AAAA query.

Default value: VAL_NOT_SET

Maximum value: 10000

Use this command to remove dns parameter settings.Refer to the set dns parameter command for meanings of the arguments.

unset dns parameter [-retries] [-minTTL] [-maxTTL] [-cacheRecords] [-nameLookupPriority] [-recursion] [-resolutionOrder] [-dnssec] [-maxPipeline] [-dnsRootReferral] [-dns64Timeout]

Displays the global DNS parameters.

show dns parameter

format

level

retries

Maximum number of retry attempts when no response is received for a query sent to a name server. Applies to end resolver and forwarder configurations.

minTTL

Minimum permissible time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is lower than the value configured for minTTL, the TTL of the record is set to the value of minTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

maxTTL

Maximum time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is higher than the value configured for maxTTL, the TTL of the record is set to the value of maxTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

nameLookupPriority

Type of lookup (DNS or WINS) to attempt first. If the first-priority lookup fails, the second-priority lookup is attempted. Used only by the SSL VPN feature.

cacheRecords

Cache resource records in the DNS cache. Applies to resource records obtained through proxy configurations only. End resolver and forwarder configurations always cache records in the DNS cache, and you cannot disable this behavior. When you disable record caching, the appliance stops caching server responses. However, cached records are not flushed. The appliance does not serve requests from the cache until record caching is enabled again.

recursion

Function as an end resolver and recursively resolve queries for domains that are not hosted on the

NetScaler appliance. Also resolve queries recursively when the external name servers configured on the appliance (for a forwarder configuration) are unavailable. When external name servers are unavailable, the appliance queries a root server and resolves the request recursively, as it does for an end resolver configuration.

resolutionOrder

Type of DNS queries (A, AAAA, or both) to generate during the routine functioning of certain NetScaler features, such as SSL VPN, cache redirection, and the integrated cache. The queries are sent to the external name servers that are configured for the forwarder function. If you specify both query types, you can also specify the order. Available settings function as follows:

* OnlyAQuery. Send queries for IPv4 address records (A records) only.

* OnlyAAAAQuery. Send queries for IPv6 address records (AAAA records) instead of queries for IPv4 address records (A records).

* AThenAAAAQuery. Send a query for an A record, and then send a query for an AAAA record if the query for the A record results in a NODATA response from the name server.

* AAAAThenAQuery. Send a query for an AAAA record, and then send a query for an A record if the query for the AAAA record results in a NODATA response from the name server.

dnssec

Enable or disable the Domain Name System Security Extensions (DNSSEC) feature on the appliance. Note: Even when the DNSSEC feature is enabled, forwarder configurations (used by internal NetScaler features such as SSL VPN and Cache Redirection for name resolution) do not support the DNSSEC OK (DO) bit in the EDNS0 OPT header.

maxPipeline

Maximum value of the concurrent DNS pipeline. A setting of zero makes the pipeline infinite

dnsRootReferral

Send a root referral if a client queries a domain name that is unrelated to the domains configured/cached on the NetScaler appliance. If the setting is disabled, the appliance sends a blank response instead of a root referral. Applicable to domains for which the appliance is authoritative. Disable the parameter when the appliance is under attack from a client that is sending a flood of queries for unrelated domains.

dns64Timeout

While doing DNS64 resolution, this parameter specifies the time to wait before sending an A query if no response is received from backend DNS server for AAAA query.

dns policy

Sep 22, 2015

The following operations can be performed on "dns policy":

add | rm | set | show

Creates a DNS policy.

add dns policy <name> <rule> <actionName>

name

Name for the DNS policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

* On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

Example: CLIENT.UDP.DNS.DOMAIN.EQ("domainname")

viewName

The view name that must be used for the given policy.

preferredLocation

The location used for the given policy. This is deprecated attribute. Please use -prefLocList

preferredLocList

The location list in priority order used for the given policy.

drop

The dns packet must be dropped.

Possible values: YES, NO

cacheBypass

By pass dns cache for this.

Possible values: YES, NO

actionName

Name of the DNS action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* dns_default_act_Drop. Drop the DNS request.

* dns_default_act_Cachebypass. Bypass the DNS cache and forward the request to the name server.

You can create custom actions by using the add dns action command in the CLI or the DNS > Actions > Create DNS Action dialog box in the NetScaler configuration utility.

add dns policy pol1 "dns.req.question.type.ne(aaaa)" -actionName act1 add dns policy pol2 "CLIENT.IP.SRC.IN_SUBNET(1.1.1.1/24)" -actionName action1 add dns policy

---

Removes a DNS policy.

rm dns policy <name>

name

Name of the DNS policy to remove.

Modifies the parameters of the specified DNS policy.

set dns policy <name> [<rule>] [-actionName <string>]

name

Name of the DNS policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

* On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

Example: CLIENT.UDP.DNS.DOMAIN.EQ("domainname")

viewName

The view name that must be used for the given policy

preferredLocation

The location used for the given policy. This is deprecated attribute. Please use -prefLocList

preferredLocList

The location list in priority order used for the given policy.

drop

The dns packet must be dropped.

Possible values: YES, NO

cacheBypass

By pass dns cache for this.

Possible values: YES, NO

actionName

Name of the DNS action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* dns_default_act_Drop. Drop the DNS request.

* dns_default_act_Cachebypass. Bypass the DNS cache and forward the request to the name server.

You can create custom actions by using the add dns action command in the CLI or the DNS > Actions > Create DNS Action dialog box in the NetScaler configuration utility.

set dns policy pol1 -rule "dns.req.question.type.ne(aaaa)" set dns policy pol2 -rule "CLIENT.IP.SRC.IN_SUBNET(1.1.1.1/24)" set dns policy pol1 -rule dns.res.header.rcod

---

Displays the parameters of the specified DNS policy or, if no policy name is specified, all configured DNS policies.

show dns policy [<name>]

name

Name of the DNS policy.

summary

fullValues

format

level

rule

The expression to be used by the dns policy.

viewName

The view name that must be used for the given policyNOTE: This attribute is deprecated.This is deprecated attribute. Please use -actionName

preferredLocation

The location used for the given policy. This is deprecated attribute. Please use -prefLocListNOTE: This attribute is deprecated.This is deprecated attribute. Please use -actionName

preferredLocList

The location list in priority order used for the given policy.NOTE: This attribute is deprecated.This is deprecated attribute. Please use -actionName

hits

The number of times the policy has been hit.

undefHits

Number of Undef hits.

drop

The dns packet must be dropped.NOTE: This attribute is deprecated.This is deprecated attribute. Please use -actionName

actionName

Name of the DNS action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* dns_default_act_Drop. Drop the DNS request.

* dns_default_act_Cachebypass. Bypass the DNS cache and forward the request to the name server.

You can create custom actions by using the add dns action command in the CLI or the DNS > Actions > Create DNS Action dialog box in the NetScaler configuration utility.

cacheBypass

By pass dns cache for this.NOTE: This attribute is deprecated.This is deprecated attribute. Please use -actionName

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

builtin

Flag to determine whether DNS policy is default or not

stateflag

devno

count

dns policy64

Sep 22, 2015

The following operations can be performed on "dns policy64":

add | rm | set | show

Creates a DNS64 Policy.

add dns policy64 <name> -rule <expression> -action <string>

name

Name for the DNS64 policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

* On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

Example: CLIENT.IP.SRC.IN_SUBENT(23.34.0.0/16)

action

Name of the DNS64 action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* A default dns64 action with prefix <default prefix> and mapped and exclude are any

You can create custom actions by using the add dns action command in the CLI or the DNS64 > Actions > Create DNS64 Action dialog box in the NetScaler configuration utility.

```
add dns64 policy pol1 "client.ip.src.in_subnet(23.43.0.0/16)" -action act1
```

Removes a DNS64 Policy.

rm dns policy64 <name>

name

Name of the DNS64 policy to be removed.

Modifies the parameters of the specified DNS64 policy.

set dns policy64 <name> [-rule <expression>] [-action <string>]

name

Name of the DNS policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

* On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

Example: CLIENT.IP.SRC.IN_SUBENT(23.34.0.0/16)

action

Name of the DNS64 action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* A default dns64 action with prefix <default prefix> and mapped and exclude are any

You can create custom actions by using the add dns action command in the CLI or the DNS64 > Actions > Create DNS64 Action dialog box in the NetScaler configuration utility.

```
set dns policy pol2 -rule "CLIENT.IP.SRC.IN_SUBNET(1.1.1.1/24)"
```

Displays the parameters of the specified DNS64 policy or, if no policy name is specified, all configured DNS64 policies.

```
show dns policy64 [<name>]
```

name

Name of the DNS64 policy.

summary

fullValues

format

level

rule

The expression to be used by the dns policy.

hits

The number of times the policy has been hit.

action

Name of the DNS64 action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* A default dns64 action with prefix <default prefix> and mapped and exclude are any

You can create custom actions by using the add dns action command in the CLI or the DNS64 > Actions > Create DNS64 Action dialog box in the NetScaler configuration utility.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

boundTo

Location where policy is bound

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

undefHits

Number of Undef hits.

description

Description of the policy

stateflag

devno

count

dns policylabel

Sep 22, 2015

The following operations can be performed on "dns policylabel":

add | rm | bind | unbind | show | stat | rename

Add a dns policy label.

add dns policylabel <labelName> <transform>

labelName

Name of the dns policy label.

transform

The type of transformations allowed by the policies bound to the label.

Possible values: dns_req, dns_res

add dns policylabel trans_dns dns_req

Remove a dns policy label.

rm dns policylabel <labelName>

labelName

Name of the dns policy label.

rm dns policylabel trans_dns

Bind the dns policy to one of the labels.

bind dns policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

labelName

Name of the dns policy label.

policyName

The dns policy name.

i) bind dns policylabel trans_dns pol_1 1 2 -invoke reqvserver CURRENT ii) bind rewrite policylabel trans_http_url pol_2 2

Unbind entities from dns label.

unbind dns policylabel <labelName> <policyName> [-priority <positive_integer>]

labelName

Name of the dns policy label.

policyName

The dns policy name.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind dns policylabel trans_dns pol_1

Display policy label or policies bound to dns policylabel.

show dns policylabel [<labelName>]

labelName

Name of the dns policy label.

summary

fullValues

format

level

stateflag

transform

The type of transformations allowed by the policies bound to the label.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

The dns policy name.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke flag.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound dns policy.

description

Description of the policylabel

isDefault

A value of true is returned if it is a default dns policylabel.

flags

devno

count

i) show dns policylabel trans_dns ii) show dns policylabel

Display statistics of dns policylabel(s).

stat dns policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

labelName

The name of the dns policy label for which statistics will be displayed. If not given statistics are shown for all dns policylabels.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy Label Hits (Hits)

Number of times policy label was invoked.

Rename a dns policy label.

rename dns policylabel <labelName>@ <newName>@

labelName

The name of the dns policylabel.

newName

The new name of the dns policylabel.

rename dns policylabel oldname newname

---

dns proxyRecords

Sep 22, 2015

The following operations can be performed on "dns proxyRecords":

Flushes all the proxy records from the DNS cache on the NetScaler appliance.

flush dns proxyRecords

dns ptrRec

Sep 22, 2015

The following operations can be performed on "dns ptrRec":

add | rm | show

Creates a pointer (PTR) record for the specified reverse domain name.

add dns ptrRec <reverseDomain> <domain> ... [-TTL <secs>]

reverseDomain

Reversed domain name representation of the IPv4 or IPv6 address for which to create the PTR record. Use the "in-addr.arpa." suffix for IPv4 addresses and the "ip6.arpa." suffix for IPv6 addresses.

domain

Domain name for which to configure reverse mapping.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

add dns ptrrec 1.1.1.in-addr.arpa. abc.com

Removes a pointer (PTR) record for the specified domain name and reverse domain name.

rm dns ptrRec <reverseDomain> [<domain> ...]

reverseDomain

Reverse domain name of the PTR record.

domain

Domain name for which to remove reverse mapping.

rm dns ptrrec 1.1.1.1.in-addr.arpa. ptr.com

Displays the pointer (PTR) record for the specified reverse domain name and domain name.

show dns ptrRec [<reverseDomain> | -type <type>]

reverseDomain

Reversed domain name representation of the IPv4 or IPv6 address for which to create the PTR record. Use the "in-addr.arpa." suffix for IPv4 addresses and the "ip6.arpa." suffix for IPv6 addresses.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

domain

Domain name for which to configure reverse mapping.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Authentication type.

devno

count

stateflag

dns records

Sep 22, 2015

The following operations can be performed on "dns records":

Displays statistics for the specified DNS record or query type. If a DNS record or query type is not specified, statistics for all record and query types are shown.

stat dns records [<dnsRecordType>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

dnsRecordType

Display statistics for the specified DNS record or query type or, if a record or query type is not specified, statistics for all record types supported on the NetScaler appliance.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Total entries (totEntries)

Total number of DNS record entries

Total updates (totUpdates)

Total number of DNS proactive updates

Total responses (totResp)

Total number of DNS server responses

Total requests (totReq)

Total number of DNS queries recieved

Current entries (curEnt)

Current number of DNS entries

Total limit errors (errLim)

Total number of times we have recieved dns record with more entries than we support

Total response format errors (errRespFor)

Total number of times we have recieved malformed responses from the backend

Total alias exist errors (errAlias)

Total number of times we have recieved non-cname record for a domain for which an alias exists

Total cache misses (errNoDom)

Total number of cache misses

Current records (curRec)

Current number of DNS Records

dns soaRec

Sep 22, 2015

The following operations can be performed on "dns soaRec":

add | rm | set | unset | show

Creates a Start of Authority (SOA) record. Note: You can set the SOA parameters that are associated with zone transfers. However, the NetScaler appliance currently does not support zone transfers.

add dns soaRec <domain> -originServer <string> -contact <string> [-serial <positive_integer>] [-refresh <secs>] [-retry <secs>] [-expire <secs>] [-minimum <secs>] [-TTL <secs>]

domain

Domain name for which to add the SOA record.

originServer

Domain name of the name server that responds authoritatively for the domain.

contact

Email address of the contact to whom domain issues can be addressed. In the email address, replace the @ sign with a period (.). For example, enter domainadmin.example.com instead of domainadmin@example.com.

serial

The secondary server uses this parameter to determine whether it requires a zone transfer from the primary server.

Default value: 100

Maximum value: 4294967294

refresh

Time, in seconds, for which a secondary server must wait between successive checks on the value of the serial number.

Default value: 3600

Maximum value: 4294967294

retry

Time, in seconds, between retries if a secondary server's attempt to contact the primary server for a zone refresh fails.

Default value: 3

Maximum value: 4294967294

expire

Time, in seconds, after which the zone data on a secondary name server can no longer be considered authoritative because all refresh and retry attempts made during the period have failed. After the expiry period, the secondary server stops serving the zone. Typically one week. Not used by the primary server.

Default value: 3600

Maximum value: 4294967294

minimum

Default time to live (TTL) for all records in the zone. Can be overridden for individual records.

Default value: 5

Maximum value: 2147483647

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Removes the Start of Authority (SOA) record for the specified domain name.

rm dns soaRec <domain>

domain

Domain name of the SOA record.

Modifies the parameters of the specified Start Of Authority (SOA) record.

set dns soaRec <domain> [-originServer <string>] [-contact <string>] [-serial <positive_integer>] [-refresh <secs>] [-retry <secs>] [-expire <secs>] [-minimum <secs>] [-TTL <secs>]

domain

Domain of the SOA record to be modified.

originServer

Domain name of the name server that responds authoritatively for the domain.

contact

Email address of the contact to whom domain issues can be addressed. In the email address, replace the @ sign with a period (.). For example, enter domainadmin.example.com instead of domainadmin@example.com.

serial

The secondary server uses this parameter to determine whether it requires a zone transfer from the primary server.

Default value: 100

Minimum value: 1

Maximum value: 4294967294

refresh

Time, in seconds, for which a secondary server must wait between successive checks on the value of the serial number.

Default value: 3600

Maximum value: 4294967294

retry

Time, in seconds, between retries if a secondary server's attempt to contact the primary server for a zone refresh fails.

Default value: 3

Maximum value: 4294967294

expire

Time, in seconds, after which the zone data on a secondary name server can no longer be considered authoritative because all refresh and retry attempts made during the period have failed. After the expiry period, the secondary server stops serving the zone. Typically one week. Not used by the primary server.

Default value: 3600

Maximum value: 4294967294

minimum

Default time to live (TTL) for all records in the zone. Can be overridden for individual records.

Default value: 5

Maximum value: 2147483647

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Use this command to remove dns soaRec settings.Refer to the set dns soaRec command for meanings of the arguments.

unset dns soaRec <domain> [-serial] [-refresh] [-retry] [-expire] [-minimum] [-TTL]

Displays the parameters of the specified Start of Authority (SOA) record. If no domain name is specified, all SOA records are displayed.

show dns soaRec [<domain> | -type <type>]

domain

The domain name.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

originServer

Domain name of the name server that responds authoritatively for the domain.

contact

Email address of the contact to whom domain issues can be addressed. In the email address, replace the @ sign with a period (.). For example, enter domainadmin.example.com instead of domainadmin@example.com.

serial

The secondary server uses this parameter to determine whether it requires a zone transfer from the primary server.

refresh

Time, in seconds, for which a secondary server must wait between successive checks on the value of the serial number.

retry

Time, in seconds, between retries if a secondary server's attempt to contact the primary server for a zone refresh fails.

expire

Time, in seconds, after which the zone data on a secondary name server can no longer be considered authoritative because all refresh and retry attempts made during the period have failed. After the expiry period, the secondary server stops serving the zone. Typically one week. Not used by the primary server.

minimum

Default time to live (TTL) for all records in the zone. Can be overridden for individual records.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns srvRec

Sep 22, 2015

The following operations can be performed on "dns srvRec":

add | rm | set | unset | show

Creates a service (SRV) record for the service offered by the specified target host, in the specified domain.

add dns srvRec <domain> <target> -priority <positive_integer> -weight <positive_integer> -port <positive_integer> [-TTL <secs>]

domain

Domain name, which, by convention, is prefixed by the symbolic name of the desired service and the symbolic name of the desired protocol, each with an underscore (_) prepended. For example, if an SRV-aware client wants to discover a SIP service that is provided over UDP, in the domain example.com, the client performs a lookup for _sip._udp.example.com.

target

Target host for the specified service.

priority

Integer specifying the priority of the target host. The lower the number, the higher the priority. If multiple target hosts have the same priority, selection is based on the Weight parameter.

Maximum value: 65535

weight

Weight for the target host. Aids host selection when two or more hosts have the same priority. A larger number indicates greater weight.

Maximum value: 65535

port

Port on which the target host listens for client requests.

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the

specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Removes, from the specified domain, the SRV record created for the service provided by the specified target host.

rm dns srvRec <domain> <target> ...

domain

Domain name of the SRV record.

target

Target host for the specified service.

Modifies the parameters of the specified service (SRV) record.

set dns srvRec <domain> <target> [-priority <positive_integer>] [-weight <positive_integer>] [-port <positive_integer>] [-TTL <secs>]

domain

Name of the SRV record to be modified.

target

Target of the SRV record to be modified.

priority

Integer specifying the priority of the target host. The lower the number, the higher the priority. If multiple target hosts have the same priority, selection is based on the Weight parameter.

Maximum value: 65535

weight

Weight for the target host. Aids host selection when two or more hosts have the same priority. A larger number indicates greater weight.

Maximum value: 65535

port

Port on which the target host listens for client requests.

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Use this command to remove dns srvRec settings.Refer to the set dns srvRec command for meanings of the arguments.

unset dns srvRec <domain> <target> -TTL

Displays the service (SRV) record configured for the specified target host and domain. If the domain name is not specified, all of the SRV records are shown.

show dns srvRec [(<domain> [<target>]) | -type <type>]

domain

Domain name for which to display the SRV record.

target

Target host for the specified service.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

priority

Priority of the target host. This helps in server selection by the client.

weight

Weight for the target host. Aids host selection when two or more hosts have the same priority. A larger number indicates greater weight.

port

Port on which the target host listens for client requests.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns stats

Sep 22, 2015

The following operations can be performed on "dns stats":

show dns stats is an alias for stat dns

show dns stats - alias for 'stat dns'

dns suffix

Sep 22, 2015

The following operations can be performed on "dns suffix":

add | rm | show

Specifies a suffix that can be used to complete domain names that are not fully qualified. For example, if you specify the example.com suffix, and the NetScaler appliance is required to resolve the incomplete domain name "myhost," it attempts to resolve "myhost.example.com."

add dns suffix <dnsSuffix>

dnsSuffix

Suffix to be appended when resolving domain names that are not fully qualified.

add dns suffix netscaler.com If the incoming domain name "engineering" is not resolved by itself, the system will append the suffix netscaler.com and attempt to resolve

---

Removes a DNS suffix.

rm dns suffix <dnsSuffix>

dnsSuffix

DNS suffix to remove.

Displays the specified DNS suffix or, if no DNS suffix is specified, all configured DNS suffixes.

show dns suffix [<dnsSuffix>]

dnsSuffix

DNS suffix to display.

summary

fullValues

format

level

devno

count

stateflag

---

dns txtRec

Sep 22, 2015

The following operations can be performed on "dns txtRec":

add | rm | show

Creates a text (TXT) record for the specified domain name. Each resource record is stored with a unique, internally generated record ID, which you can view and use to delete the record. You cannot modify a TXT resource record.

add dns txtRec <domain> <string> ... [-TTL <secs>]

domain

Name of the domain for the TXT record.

string

Information to store in the TXT resource record. Enclose the string in single or double quotation marks. A TXT resource record can contain up to six strings, each of which can contain up to 255 characters. If you want to add a string of more than 255 characters, evaluate whether splitting it into two or more smaller strings, subject to the six-string limit, works for you.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

add dns txtRec spf.m.test. "v=spf1 ip4:1.2.3.0/24 ip4:1.3.4.0/24 ?all" add dns txtRec comments.m.test. "This is a CHARSTR" "This is another CHARSTR"

Removes the specified TXT record from the specified domain.

rm dns txtRec <domain> (<string> ... | -recordId <positive_integer>@)

domain

Name of the domain for the TXT record.

string

Complete set of text strings in the TXT record, entered in the order in which they are stored in the record. Mutually exclusive with the record ID parameter.

recordId

Unique, internally generated record ID. View the details of the TXT record to obtain its record ID. Mutually exclusive with the string parameter.

Minimum value: 1

Maximum value: 65535

rm dns txtRec spf.m.test. "v=spf1 ip4:1.2.3.0/24 ip4:1.3.4.0/24 ?all" rm dns txtRec comments.m.test. "This is a CHARSTR" "This is another CHARSTR" rm dns txtRec co

---

Displays TXT records owned by the specified domain. If no domain name is specified, all configured TXT records are shown.

show dns txtRec [<domain> | -type <type>]

domain

Name of the domain for the TXT record.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: NSDNS_AUTH_HOST

summary

fullValues

format

level

string

Information to store in the TXT resource record. Enclose the string in single or double quotation marks. A TXT resource record can contain up to six strings, each of which can contain up to 255 characters. If you want to add a string of more than 255 characters, evaluate whether splitting it into two or more smaller strings, subject to the six-string limit, works for you.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

recordId

authType

Authentication type.

devno

count

stateflag

show dns txtRec spf.m.test. show dns txtRec

dns view

Sep 22, 2015

The following operations can be performed on "dns view":

add | rm | show

Creates a DNS view. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

add dns view <viewName>

viewName

Name for the DNS view.

add dns view privateview

Removes a DNS view.

rm dns view <viewName>

viewName

Name for the DNS view.

rm dns view privateview

Displays the specified DNS view or, if no DNS view name is specified, all the DNS views configured on the NetScaler appliance.

show dns view [<viewName>]

viewName

Name of the view to display.

summary

fullValues

format

level

serviceName

Service name of the service using this view.NOTE: This attribute is deprecated.This attribute is deprecated. please use -gslbservicename

gslbServiceName

Service name of the service using this view.

dnsPolicyName

dnspolicy name of this view.

IPAddress

IP of the service corresponding to the given view.

flags

Flags controlling display.NOTE: This attribute is deprecated.This is deprecated attribute.

stateflag

flags controlling display

devno

count

dns zone

Sep 22, 2015

The following operations can be performed on "dns zone":

add | set | unset | rm | sign | unsign | show

Creates a DNS zone on the NetScaler appliance. Mandatory if you want to use the appliance to implement Domain Name Security Extensions (DNSSEC) for the zone. When you add a DNS resource record, if the domain name of the record belongs to the zone, the record is automatically added to the zone.

add dns zone <zoneName> -proxyMode (YES | NO) [-dnssecOffload (ENABLED | DISABLED) [-nsec (ENABLED | DISABLED)]]

zoneName

Name of the zone to create.

proxyMode

Deploy the zone in proxy mode. Enable in the following scenarios:

* The load balanced DNS servers are authoritative for the zone and all resource records that are part of the zone.

* The load balanced DNS servers are authoritative for the zone, but the NetScaler appliance owns a subset of the resource records that belong to the zone (partial zone ownership configuration). Typically seen in global server load balancing (GSLB) configurations, in which the appliance responds authoritatively to queries for GSLB domain names but forwards queries for other domain names in the zone to the load balanced servers.

In either scenario, do not create the zone's Start of Authority (SOA) and name server (NS) resource records on the appliance.

Disable if the appliance is authoritative for the zone, but make sure that you have created the SOA and NS records on the appliance before you create the zone.

Possible values: YES, NO

Default value: ENABLED

add dns zone foo.bar -proxyMode NO -dnssec ENABLED

Modifies the parameters of the specified DNS zone.

set dns zone <zoneName> [-proxyMode (YES | NO)] [-dnssecOffload (ENABLED | DISABLED)] [-nsec (ENABLED | DISABLED)]

zoneName

Name of the zone.

proxyMode

Deploy the zone in proxy mode. Enable in the following scenarios:

* The load balanced DNS servers are authoritative for the zone and all resource records that are part of the zone.

* The load balanced DNS servers are authoritative for the zone, but the NetScaler appliance owns a subset of the resource records that belong to the zone (partial zone ownership configuration). Typically seen in global server load balancing (GSLB) configurations, in which the appliance responds authoritatively to queries for GSLB domain names but forwards queries for other domain names in the zone to the load balanced servers.

In either scenario, do not create the zone's Start of Authority (SOA) and name server (NS) resource records on the appliance.

Disable if the appliance is authoritative for the zone, but make sure that you have created the SOA and NS records on the appliance before you create the zone.

Possible values: YES, NO

Default value: ENABLED

set dns zone foo.bar -proxyMode NO -dnssec ENABLED

Use this command to remove dns zone settings.Refer to the set dns zone command for meanings of the arguments.

unset dns zone <zoneName> [-proxyMode] [-dnssecOffload] [-nsec]

Removes a DNS zone from the NetScaler appliance.

rm dns zone <zoneName>

zoneName

Name of the zone to remove.

Signs a DNS zone with a DNS key. Before you sign a zone, make sure that you've enabled DNSSEC by setting the global DNS parameter "Enable DNSSEC extension."

sign dns zone <zoneName> [-keyName <string> ...]

zoneName

Name of the zone.

keyName

Name of the public/private DNS key pair with which to sign the zone. You can sign a zone with up to four keys.

sign dns zone abc.com. -keyname abc.com.zsk abc.com.ksk

Unsigns the specified DNS zone with the specified DNS key.

unsign dns zone <zoneName> [-keyName <string> ...]

zoneName

Name of the zone.

keyName

Name of the public-private DNS key pair with which to unsign the zone.

unsign dns zone abc.com. -keyname abc.com.zsk abc.com.ksk

Displays the parameters of the specified DNS zone, along with information about the types of resource records available for each domain name in the zone. If no zone name is specified, just the parameters are shown, for all configured zones.

show dns zone [<zoneName> | -type <type>]

zoneName

Name of the zone. Mutually exclusive with the type parameter.

type

Type of zone to display. Mutually exclusive with the DNS Zone (zoneName) parameter. Available settings function as follows:

* ADNS - Display all the zones for which the NetScaler appliance is authoritative.

* PROXY - Display all the zones for which the NetScaler appliance is functioning as a proxy server.

* ALL - Display all the zones configured on the appliance.

Possible values: ALL, ADNS, PROXY

summary

fullValues

format

level

proxyMode

Deploy the zone in proxy mode. Enable in the following scenarios:

* The load balanced DNS servers are authoritative for the zone and all resource records that are part of the zone.

* The load balanced DNS servers are authoritative for the zone, but the NetScaler appliance owns a subset of the resource records that belong to the zone (partial zone ownership configuration). Typically seen in global server load balancing (GSLB) configurations, in which the appliance responds authoritatively to queries for GSLB domain names but forwards queries for other domain names in the zone to the load balanced servers.

In either scenario, do not create the zone's Start of Authority (SOA) and name server (NS)

resource records on the appliance.

Disable if the appliance is authoritative for the zone, but make sure that you have created the SOA and NS records on the appliance before you create the zone.

flags

Flags controlling display.

nsecBitarray

Bit array representing the different record types configured for the domain nameNOTE: This attribute is deprecated.This is deprecated attribute.

domain

Domain name that belongs to the given zone

nextRecs

An array of record types associated with the nsec record.

stateflag

flags controlling display

dnssecOffload

Enable dnssec offload for this zone.

nsec

Enable nsec generation for dnssec offload.

devno

count

show dns zone foo.bar

DOS Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- dos
- dos policy
- dos stats

dos

Sep 22, 2015

The following operations can be performed on "dos":

Displays DoS protection statistics.

stat dos [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

DOS condition triggered (CndMatch)

Number of times the NetScaler appliance triggered the DOS JavaScript due to a condition match.

Valid DOS clients (ValidClt)

Number of clients from whom the NetScaler appliance received a valid DOS cookie.

DOS priority clients (DosPriCl)

Number of valid clients that were given DOS priority.

dos policy

Sep 22, 2015

The following operations can be performed on "dos policy":

add | rm | set | unset | show | stat

Adds a DoS protection policy to the appliance. Note: To apply DoS protection to a service, bind the DoS policy to the service by using the bind service command.

add dos policy <name> -qDepth <positive_integer> [-cltDetectRate <positive_integer>]

name

Name for the HTTP DoS protection policy. Must begin with a letter, number, or the underscore character (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), and colon (:) characters.

qDepth

Queue depth. The queue size (the number of outstanding service requests on the system) before DoS protection is activated on the service to which the DoS protection policy is bound.

Minimum value: 21

cltDetectRate

Client detect rate. Integer representing the percentage of traffic to which the HTTP DoS policy is to be applied after the queue depth condition is satisfied.

Maximum value: 100

add dos policy dospol -qdepth 100 -cltDetectRate 90

Removes a DoS protection policy from the appliance.

rm dos policy <name>

name

Name of the DoS protection policy to be removed.

rm dos policy dospol

Modifies the attributes of a DoS protection policy.

set dos policy <name> [-qDepth <positive_integer>] [-cltDetectRate <positive_integer>]

name

Name of the DoS protection policy to be modified.

qDepth

Queue depth. The queue size (the number of outstanding service requests on the system) before DoS protection is activated on the service to which the DoS

protection policy is bound.

Minimum value: 21

cltDetectRate

Client detect rate. Integer representing the percentage of traffic to which the HTTP DoS policy is to be applied after the queue depth condition is satisfied.

Minimum value: 1

Maximum value: 100

```
set dos policy dospol -qdepth 1000
```

Use this command to remove dos policy settings.Refer to the set dos policy command for meanings of the arguments.

```
unset dos policy <name> -cltDetectRate
```

Displays information about a DoS protection policy.

```
show dos policy [<name>]
```

name

Name of the DoS protection policy about which to display information. If a name is not provided, information about all DoS protection policies is shown.

summary

fullValues

format

level

qDepth

Queue depth. The queue size (the number of outstanding service requests on the system) before DoS protection is activated on the service to which the DoS protection policy is bound.

cltDetectRate

Client detect rate. Integer representing the percentage of traffic to which the HTTP DoS policy is to be applied after the queue depth condition is satisfied.

devno

count

stateflag

```
> show dos policy       1 configured DoS policy: 1)     Policy: dospol  QDepth: 100     ClientDetectRate: 90  Done
```

Displays statistics of the DoS protection policy.

stat dos policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

The name of the DoS protection policy whose statistics must be displayed. If a name is not provided, statistics of all the DoS protection policies are displayed.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Client detect rate (ClDtRate)

Current ratio of JavaScript send rate to the server response rate (Client detect rate)

Physical service IP (SvcIP)

IP address of the service to which this policy is bound.

Physical service port (SvcPort)

Port address of the service to which this policy is bound.

Current server queue size (CurQSize)

Current queue size of the server to which this policy is bound.

DOS transactions (DosTrans)

Total number of DoS JavaScript transactions performed for this policy.

Client detect rate mismatch (JsRefusd)

Number of times the DoS JavaScript was not sent because the set JavaScript rate was not met for this policy.

Valid clients (TotValCl)

Total number of valid DoS cookies received for this policy.

DOS JavaScript bytes served (JsBytSnt)

Total number of DoS JavaScript bytes sent for this policy.

Non GET, POST requests

Number of non-GET and non-POST requests for which DOS JavaScript was sent.

DOS JavaScript send rate (JSRate)

Current rate at which JavaScript is being sent in response to client requests.

Server response rate (RespRate)

Current rate at which the server to which this policy is bound is responding.

dos stats

Sep 22, 2015

The following operations can be performed on "dos stats":

show dos stats is an alias for stat dos Displays DoS protection statistics.

show dos stats - alias for 'stat dos'

Filter Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- filter action
- filter global
- filter htmlinjectionparameter
- filter htmlinjectionvariable
- filter policy
- filter postbodyInjection
- filter prebodyInjection

filter action

Sep 22, 2015

The following operations can be performed on "filter action":

add | rm | set | unset | show

Creates a content filtering action. This action can be associated with a content filtering policy that is created with the add filter policy command. Note: The following content filtering actions are available by default: * RESET - Sends a TCP reset for the HTTP requests. * DROP - Drops the HTTP requests silently, without sending a TCP FIN for closing the connection.

add filter action <name> <qual> [<serviceName>] [<value>] [<respCode>] [<page>]

name

Name for the filtering action. Must begin with a letter, number, or the underscore character (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space ( ), at sign (@), equals (=), and colon (:) characters. Choose a name that helps identify the type of action. The name of a filter action cannot be changed after it is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

qual

Qualifier, which is the action to be performed. The qualifier cannot be changed after it is set. The available options function as follows:

ADD - Adds the specified HTTP header.

RESET - Terminates the connection, sending the appropriate termination notice to the user's browser.

FORWARD - Redirects the request to the designated service. You must specify either a service name or a page, but not both.

DROP - Silently deletes the request, without sending a response to the user's browser.

CORRUPT - Modifies the designated HTTP header to prevent it from performing the function it was intended to perform, then sends the request/response to the server/browser.

ERRORCODE. Returns the designated HTTP error code to the user's browser (for example, 404, the standard HTTP code for a non-existent Web page).

Possible values: reset, add, corrupt, forward, errorcode, drop

serviceName

Service to which to forward HTTP requests. Required if the qualifier is FORWARD.

value

String containing the header_name and header_value. If the qualifier is ADD, specify <header_name>:<header_value>. If the qualifier is CORRUPT, specify only the header_name

respCode

Response code to be returned for HTTP requests (for use with the ERRORCODE qualifier).

Minimum value: 1

page

HTML page to return for HTTP requests (For use with the ERRORCODE qualifier).

add filter action bad_url_action errorcode 400 "<HTML>Bad URL.</HTML>" add filter action forw_action FORWARD service1 add filter action add_header_action add "HI

Removes a content filtering action.

rm filter action <name>

name

Name of the content filter action to be removed.

rm filter action filter_action_name

Modifies an existing content filtering action.

set filter action <name> [-serviceName <string>] [-value <string>] [-respCode <positive_integer>] [-page <string>]

name

Name of the content filtering action to be modified.

serviceName

Service to which to forward HTTP requests. Required if the qualifier is FORWARD.

value

String containing the header_name and header_value. If the qualifier is ADD, specify <header_name>:<header_value>. If the qualifier is CORRUPT, specify only the header_name

respCode

Response code to be returned for HTTP requests (for use with the ERRORCODE qualifier).

Minimum value: 1

page

HTML page to return for HTTP requests (For use with the ERRORCODE qualifier).

set filter action bad_url_action -respcode 400 -page "<HTML>Bad URL.</HTML>" set filter action forw_action -serviceName service1 set filter action add_header_action

---

Use this command to remove filter action settings.Refer to the set filter action command for meanings of the arguments.

unset filter action <name> -page

Displays information about available filtering actions.

show filter action [<name>]

name

Name of the content filtering action to be displayed. If a name is not provided, information about all filter actions is shown.

summary

fullValues

format

level

qual

Qualifier, which is the action to be performed. The qualifier cannot be changed after it is set. The available options function as follows:

ADD - Adds the specified HTTP header.

RESET - Terminates the connection, sending the appropriate termination notice to the user's browser.

FORWARD - Redirects the request to the designated service. You must specify either a service name or a page, but not both.

DROP - Silently deletes the request, without sending a response to the user's browser.

CORRUPT - Modifies the designated HTTP header to prevent it from performing the function it was intended to perform, then sends the request/response to the server/browser.

ERRORCODE. Returns the designated HTTP error code to the user's browser (for example, 404, the standard HTTP code for a non-existent Web page).

serviceName

The service to which HTTP requests are forwarded. This parameter will exist when the qualifier is FORWARD.

value

The string containing the header_name and header_value. When the qualifier is ADD it will have header_name:header_value. When the qualifier is Corrupt this will have header_name.

respCode

The response code to be returned for HTTP requests. This parameter will exist when the qualifier is ERRORCODE.

page

The HTML page that will be returned for the HTTP requests. This parameter will exist when the qualifier is ERRORCODE.

stateflag

isDefault

A value of true is returned if it is a default filteraction.

flag

builtin

devno

count

Example 1 The following shows an example of the output of the show filter action command when no filter actions have

filter global

Sep 22, 2015

The following operations can be performed on "filter global":

bind | unbind | show

Apply (bind) the specified filtering policy globally. Note: Filtering requires the content filtering license.

bind filter global (<policyName> [-priority <positive_integer>]) [-state (ENABLED | DISABLED)]

policyName

Name of the filtering policy to be bound.

To send RESET for all the HTTP requests which are not get or head type, following filter policy can be created: add filter policy reset_invalid_req -rule "METHOD != GET

Deactivate a globally bound filter policy.

unbind filter global <policyName>

policyName

Name of the filter policy to be unbound.

Globally active filter policies can be seen using command: show filter global 1) Policy Name: reset_invalid_req Priority: 0 Done This globally active filter policy can b

---

Displays the globally activated filter policies.

show filter global

summary

fullValues

format

level

policyName

The name of the filter policy.

priority

The priority of the policy.

state

State of the binding.

stateflag

devno

count

show filter global 1) Policy Name: url_filter Priority: 0 2) Policy Name: reset_invalid_req Priority: 0 Done

filter htmlinjectionparameter

Sep 22, 2015

The following operations can be performed on "filter htmlinjectionparameter":

set | unset | show

Sets the HTML injection parameters.

set filter htmlinjectionparameter [-rate <positive_integer>] [-frequency <positive_integer>] [-strict (ENABLED | DISABLED)] [-htmlsearchlen <positive_integer>]

rate

For a rate of x, HTML injection is done for 1 out of x policy matches.

Default value: 1

Minimum value: 1

frequency

For a frequency of x, HTML injection is done at least once per x milliseconds.

Default value: 1

Minimum value: 1

strict

Searching for <html> tag. If this parameter is enabled, HTML injection does not insert the prebody or postbody content unless the <html> tag is found.

Possible values: ENABLED, DISABLED

Default value: ENABLED

htmlsearchlen

Number of characters, in the HTTP body, in which to search for the <html> tag if strict mode is set.

Default value: 1024

Minimum value: 1

set htmlinjection parameter -rate 10 -frequency 1

Removes the HTML injection settings..Refer to the set filter htmlinjectionparameter command for meanings of the arguments.

unset filter htmlinjectionparameter [-rate] [-frequency] [-strict] [-htmlsearchlen]

a) unset htmlinjectionparameter -rate b) unset htmlinjectionparameter -frequency c) unset htmlinjectionparameter -rate -frequency

Displays the HTML injection parameters.

show filter htmlinjectionparameter

format

level

rate

For a rate of x, HTML injection is done for 1 out of x policy matches.

frequency

For a frequency of x, HTML injection is done at least once per x milliseconds.

strict

Searching for <html> tag. If this parameter is enabled, HTML injection does not insert the prebody or postbody content unless the <html> tag is found.

htmlsearchlen

Number of characters, in the HTTP body, in which to search for the <html> tag if strict mode is set.

rate : 10

filter htmlinjectionvariable

Sep 22, 2015

The following operations can be performed on "filter htmlinjectionvariable":

add | rm | set | unset | show

Creates an HTML injection variable.

add filter htmlinjectionvariable <variable> [-value <string>]

variable

Name for the HTML injection variable to be added.

value

Value to be assigned to the new variable.

varId

ID of the system variable. Used only in builtins.

Possible values: IID, UTIME, XID, PAGEID, REQRTBEG, REQRTEND, REQSTBEG, REQSTEND, RESRTBEG, RESRTEND, RESSTBEG, RESSTEND, CLTRTT, CTYPE, TRANSID, SYSVSVR, SYSSERV

add htmlinjectionvariable EDGESIGHT_SERVER_IP -value 1.1.1.1

Removes an HTML injection variable.

rm filter htmlinjectionvariable <variable>

variable

Name of the HTML injection variable to be removed.

rm htmlinjectionvariable EDGESIGHT_SERVER_IP

Modifies the value of an HTML injection variable.

set filter htmlinjectionvariable <variable> [-value <string>]

variable

Name of the HTML injection variable to be modified.

value

Value to be assigned to the new variable.

set htmlinjectionvariable EDGESIGHT_SERVER_IP -value 2.2.2.2

Use this command to remove filter htmlinjectionvariable settings.Refer to the set filter htmlinjectionvariable command for meanings of the arguments.

unset filter htmlinjectionvariable <variable> -value

Displays information about HTML injection variables.

show filter htmlinjectionvariable [<variable>]

variable

Name of the HTML injection variable to be displayed. If a name is not provided, information about all the HTML injection variables is shown.

summary

fullValues

format

level

value

Value of the HTML injection variable

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

type

Type of the HTML injection variable

devno

count

stateflag

show htmlinjectionvariable EDGESIGHT_SERVER_IP

filter policy

Sep 22, 2015

The following operations can be performed on "filter policy":

add | rm | set | show

Creates a content filtering policy.

add filter policy <name> -rule <expression> (-reqAction <string> | -resAction <string>)

name

Name for the filtering action. Must begin with a letter, number, or the underscore character (_). Other characters allowed, after the first character, are the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), and colon (:) characters. Choose a name that helps identify the type of action. The name cannot be updated after the policy is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

NetScaler classic expression specifying the type of connections that match this policy.

reqAction

Name of the action to be performed on requests that match the policy. Cannot be specified if the rule includes condition to be evaluated for responses.

resAction

The action to be performed on the response. The string value can be a filter action created filter action or a built-in action.

Example 1: add policy expression e1 "sourceip == 66.33.22.0 -netmask 255.255.255.0" add policy expression e2 "URL == /admin/account.asp" add filter policy ip_filter

---

Removes a filter policy.

rm filter policy <name>

name

Name of the filter policy to be removed.

rm filter policy filter_policy_name The "show filter policy" command shows all filter policies that are currently defined.

Modifies a filter policy.

set filter policy <name> [-rule <expression>] [-reqAction <string> | -resAction <string>]

name

Name of the filter policy to be modified.

rule

NetScaler classic expression specifying the type of connections that match this policy.

reqAction

Name of the action to be performed on requests that match the policy. Cannot be specified if the rule includes condition to be evaluated for responses.

resAction

The action to be performed on the response. The string value can be a filter action created filter action or a built-in action.

Example 1: A filter policy to allow access of URL /foo/secure.asp only from 65.186.55.0 network can be created using below command: add filter policy url_filter -rule "URI

Displays information about the filter policies.

show filter policy [<name>]

name

Name of the filter policy to be displayed. If a name is not provided, information about all the filter policies is shown.

summary

fullValues

format

level

rule

NetScaler classic expression specifying the type of connections that match this policy.

reqAction

The name of the action to be performed on the request.

resAction

The action to be performed on the response.

hits

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

show filter policy 1) Name: nimda_filter Rule: (URL CONTAINS root.exe || URL CONTAINS cmd.e

---

filter postbodyInjection

Sep 22, 2015

The following operations can be performed on "filter postbodyInjection":

set | unset | show

Specifies the file to be used for postbody injection.

set filter postbodyInjection <postbody>

postbody

Name of file whose contents are to be inserted after the response body.

set filter postbodyInjection ens/postbody.js

Removes the setting that specifies the file used for postbody injection..Refer to the set filter postbodyInjection command for meanings of the arguments.

unset filter postbodyInjection [-postbody]

unset filter postbodyInjection

Displays the name of the file used for postbody injection.

show filter postbodyInjection

format

level

postbody

The name of the postbody file.

systemIID

The system IID of the current NetScaler system.

filter prebodyInjection

Sep 22, 2015

The following operations can be performed on "filter prebodyInjection":

set | unset | show

Specifies the file to be used for prebody injection.

set filter prebodyInjection <prebody>

prebody

Name of file whose contents are to be inserted before the response body.

set filter prebodyInjection ens/prebody.js

Removes the setting that specifies the file used for prebody injection..Refer to the set filter prebodyInjection command for meanings of the arguments.

unset filter prebodyInjection [-prebody]

unset filter prebodyInjection

Displays the name of the file used for prebody injection.

show filter prebodyInjection

format

level

prebody

The name of the prebody file.

systemIID

The system IID of the current NetScaler system.

GSLB Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- gslb action
- gslb config
- gslb domain
- gslb ldnsentries
- gslb ldnsentry
- gslb parameter
- gslb policy
- gslb runningConfig
- gslb service
- gslb site
- gslb syncStatus
- gslb vserver

gslb action

Sep 22, 2015

The following operations can be performed on "gslb action":

add | rm | set | show

Add GSLB action used in the GSLB policy NOTE: This command is deprecated.

name

The name of the GSLB action

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard `*' is accepted as a valid qualifier token.

```
add gslb action pref_site -preferredlocation NorthAmerica.US.*.*.*.*
```

Remove the gslb action configured in the system NOTE: This command is deprecated.

name

The name of the action to be removed

```
rm gslb action redirect_asia
```

Change the preferredlocation of the given gslb action NOTE: This command is deprecated.

name

The name of the GSLB action

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard `*' is accepted as a valid qualifier token.

set gslb action pref_site -preferredlocation NorthAmerica.US.*.*.*.*

Display the GSLB actions configured NOTE: This command is deprecated.

name

The name of the action.

format

level

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard `*' is accepted as a valid qualifier token.

devno

count

stateflag

show gslb action

---

gslb config

Sep 22, 2015

The following operations can be performed on "gslb config":

Synchronizes the GSLB running configuration on all NetScaler appliances participating in the GSLB setup. The appliance on which this command is run is considered the master node. All GSLB sites configured on the master node and not having a parent site are synchronized with the master node.

sync gslb config [-preview | -forceSync <string> | -command <string> | -nowarn | -saveconfig] [-debug]

preview

Do not synchronize the GSLB sites, but display the commands that would be applied on the slave node upon synchronization. Mutually exclusive with the Save Configuration option.

debug

Generate verbose output when synchronizing the GSLB sites. The Debug option generates more verbose output than the sync gslb config command in which the option is not used, and is useful for analyzing synchronization issues.

forceSync

Force synchronization of the specified site even if a dependent configuration on the remote site is preventing synchronization or if one or more GSLB entities on the remote site have the same name but are of a different type. You can specify either the name of the remote site that you want to synchronize with the local site, or you can specify All Sites in the configuration utility (the string all-sites in the CLI). If you specify All Sites, all the sites in the GSLB setup are synchronized with the site on the master node.

Note: If you select the Force Sync option, the synchronization starts without displaying the commands that are going to be executed.

nowarn

Suppress the warning and the confirmation prompt that are displayed before site synchronization begins. This option can be used in automation scripts that must not be interrupted by a prompt.

saveconfig

Save the configuration on all the nodes participating in the synchronization process, automatically. The master saves its configuration immediately before synchronization begins. Slave nodes save their configurations after the process of synchronization is complete. A slave node saves its configuration only if the configuration difference was successfully applied to it. Mutually exclusive with the Preview option.

command

Run the specified command on the master node and then on all the slave nodes. You cannot use this option with the force sync and preview options.

sync gslb config

gslb domain

Sep 22, 2015

The following operations can be performed on "gslb domain":

Displays the statistics associated with a global server load balancing (GSLB) domain.

stat gslb domain [<name> [-dnsRecordType <dnsRecordType>]] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the GSLB domain for which to display statistics. If you do not specify a name, statistics are shown for all configured GSLB domains.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

domainHits (Hits)

Total number of DNS queries received.

Dns Record Type (Rec_Type)

Type of DNS record returned

gslb ldnsentries

Sep 22, 2015

The following operations can be performed on "gslb ldnsentries":

clear | show

Clears all the local DNS (LDNS) entries created on the NetScaler appliance. LDNS entries store network metrics for RTT learned from the packets exchanged with LDNS servers.

clear gslb ldnsentries

Displays the local DNS (LDNS) entries created on the NetScaler appliance. LDNS entries store network metrics for RTT learned from the packets exchanged with LDNS servers.

show gslb ldnsentries

summary

fullValues

siteName

The GSLB site name

numsites

Specifies the number of gslb sites

IPAddress

IP address of the LDNS server

TTL

TTL value of the LDNS entry

name

Monitor that is currently being used to monitor the LDNS ip..

rtt

RTT value of the LDNS entry for all gslb sites

devno

count

stateflag

show gslb ldnsentries

gslb ldnsentry

Sep 22, 2015

The following operations can be performed on "gslb ldnsentry":

Removes the LDNS entry for the specified LDNS IP address.

rm gslb ldnsentry <IPAddress>

IPAddress

IP address of the LDNS server.

rm gslb ldnsentry 10.102.27.226

gslb parameter

Sep 22, 2015

The following operations can be performed on "gslb parameter":

set | unset | show

Sets various global GSLB parameters.

set gslb parameter [-ldnsEntryTimeout <secs>] [-RTTTolerance <msecs>] [-ldnsMask <netmask>] [-v6ldnsmasklen <positive_integer>] [-ldnsProbeOrder <ldnsProbeOrder> ...] [-dropLdnsReq (ENABLED | DISABLED)]

ldnsEntryTimeout

Time, in seconds, after which an inactive LDNS entry is removed.

Default value: 180

Maximum value: 65534

RTTTolerance

Tolerance, in milliseconds, for newly learned round-trip time (RTT) values. If the difference between the old RTT value and the newly computed RTT value is less than or equal to the specified tolerance value, the LDNS entry in the network metric table is not updated with the new RTT value. Prevents the exchange of metrics when variations in RTT values are negligible.

Default value: 5

Minimum value: 1

Maximum value: 100

ldnsMask

The IPv4 network mask with which to create LDNS entries.

Default value: 0xFFFFFFFF

v6ldnsmasklen

Mask for creating LDNS entries for IPv6 source addresses. The mask is defined as the number of leading bits to consider, in the source IP address, when creating an LDNS entry.

Default value: 128

Minimum value: 1

Maximum value: 128

ldnsProbeOrder

Order in which monitors should be initiated to calculate RTT.

Possible values: PING, DNS, TCP

Default value: ARRAY(0x97d6208)

dropLdnsReq

Drop LDNS requests if round-trip time (RTT) information is not available.

Possible values: ENABLED, DISABLED

Default value: DISABLED

set gslb parameter -ldnsMask 255.255.0.0

Use this command to remove gslb parameter settings.Refer to the set gslb parameter command for meanings of the arguments.

unset gslb parameter [-ldnsEntryTimeout] [-RTTTolerance] [-ldnsMask] [-v6ldnsmasklen] [-ldnsProbeOrder] [-dropLdnsReq]

Displays the global GSLB parameters.

show gslb parameter

format

level

flags

State of the GSLB parameter.

ldnsEntryTimeout

Time, in seconds, after which an inactive LDNS entry is removed.

RTTTolerance

Tolerance, in milliseconds, for newly learned round-trip time (RTT) values. If the difference between the old RTT value and the newly computed RTT value is less than or equal to the specified tolerance value, the LDNS entry in the network metric table is not updated with the new RTT value. Prevents the exchange of metrics when variations in RTT values are negligible.

ldnsMask

The IPv4 network mask with which to create LDNS entries.

v6ldnsmasklen

Mask for creating LDNS entries for IPv6 source addresses. The mask is defined as the number of leading bits to consider, in the source IP address, when creating an LDNS entry.

ldnsProbeOrder

The order in which monitors should be initiated to calculate RTT

dropLdnsReq

Drop LDNS requests if round-trip time (RTT) information is not available.

show gslb parameter

gslb policy

Sep 22, 2015

The following operations can be performed on "gslb policy":

add | rm | set | show

Add GSLB policy NOTE: This command is deprecated.

name

The name of the GSLB policy

reqRule

The expression rule

action

The GSLB action to be used when the reqrule is matched

```
add gslb policy gslb_redirect -reqRule client_Japan -action pref_site
```

Remove the gslb policy configured in the system NOTE: This command is deprecated.

name

The name of the policy to be removed

```
rm gslb policy gslb_redirect
```

Change the action for the given gslb policy NOTE: This command is deprecated.

name

The name of the gslb policy.

action

The action to be taken for the given gslb policy

set gslb policy gslb_redirect -action redirect_asia

Display the configured GSLB policy NOTE: This command is deprecated.

name

The name of the GSLB policy.

format

level

reqRule

The expression rule

action

The action taken for the given gslb policy.

hits

Number of policy hits for the gslb policy.

devno

count

stateflag

show gslb policy

gslb runningConfig

Sep 22, 2015

The following operations can be performed on "gslb runningConfig":

Displays the complete GSLB configuration running on the NetScaler appliance. In addition to the saved configuration, the running configuration includes GSLB settings that have not yet been saved to the NetScaler configuration file (ns.conf).

show gslb runningConfig

response

gslb sync status as text blob

gslb service

Sep 22, 2015

The following operations can be performed on "gslb service":

add | rm | set | unset | bind | unbind | show | stat | rename

Creates a global server load balancing (GSLB) service.

add gslb service <serviceName> [-cnameEntry <string> | <IP> | <serverName> | <serviceType> | <port> | -publicIP <ip_addr|ipv6_addr|*> | -publicPort <port> | -sitePersistence <sitePersistence> | -sitePrefix <string>] [-maxClient <positive_integer>] [-healthMonitor (YES | NO)] [-siteName <string>] [-state (ENABLED | DISABLED)] [-cip (ENABLED | DISABLED) [<cipHeader>]] [-cookieTimeout <mins>] [-cltTimeout <secs>] [-svrTimeout <secs>] [-maxBandwidth <positive_integer>] [-downStateFlush (ENABLED | DISABLED)] [-maxAAAUsers <positive_integer>] [-monThreshold <positive_integer>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)]

serviceName

Name for the GSLB service. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the GSLB service is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my gslbsvc" or 'my gslbsvc').

cnameEntry

Canonical name of the GSLB service. Used in CNAME-based GSLB.

IP

IP address for the GSLB service. Should represent a load balancing, content switching, or VPN virtual server on the NetScaler appliance, or the IP address of another load balancing device.

serverName

Name of the server hosting the GSLB service.

serviceType

Type of service to create.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, NNTP, ANY, SIP_UDP, RADIUS, RDP, RTSP, MYSQL, MSSQL

Default value: NSSVC_SERVICE_UNKNOWN

port

Port on which the load balancing entity represented by this GSLB service listens.

Minimum value: 1

publicIP

The public IP address that a NAT device translates to the GSLB service's private IP address. Optional.

publicPort

The public port associated with the GSLB service's public IP address. The port is mapped to the service's private port number. Applicable to the local GSLB service. Optional.

maxClient

The maximum number of open connections that the service can support at any given time. A GSLB service whose connection count reaches the maximum is not considered when a GSLB decision is made, until the connection count drops below the maximum.

Maximum value: 4294967294

healthMonitor

Monitor the health of the GSLB service.

Possible values: YES, NO

Default value: YES

siteName

Name of the GSLB site to which the service belongs.

state

Enable or disable the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

cip

In the request that is forwarded to the GSLB service, insert a header that stores the client's IP address. Client IP header insertion is used in connection-proxy based site persistence.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipHeader

Name for the HTTP header that stores the client's IP address. Used with the Client IP option. If client IP header

insertion is enabled on the service and a name is not specified for the header, the NetScaler appliance uses the name specified by the cipHeader parameter in the set ns param command or, in the GUI, the Client IP Header parameter in the Configure HTTP Parameters dialog box.

sitePersistence

Use cookie-based site persistence. Applicable only to HTTP and SSL GSLB services.

Possible values: ConnectionProxy, HTTPRedirect, NONE

cookieTimeout

Timeout value, in minutes, for the cookie, when cookie based site persistence is enabled.

Maximum value: 1440

sitePrefix

The site's prefix string. When the service is bound to a GSLB virtual server, a GSLB site domain is generated internally for each bound service-domain pair by concatenating the site prefix of the service and the name of the domain. If the special string NONE is specified, the site-prefix string is unset. When implementing HTTP redirect site persistence, the NetScaler appliance redirects GSLB requests to GSLB services by using their site domains.

cltTimeout

Idle time, in seconds, after which a client connection is terminated. Applicable if connection proxy based site persistence is used.

Maximum value: 31536000

svrTimeout

Idle time, in seconds, after which a server connection is terminated. Applicable if connection proxy based site persistence is used.

Maximum value: 31536000

maxBandwidth

Integer specifying the maximum bandwidth allowed for the service. A GSLB service whose bandwidth reaches the maximum is not considered when a GSLB decision is made, until its bandwidth consumption drops below the maximum.

downStateFlush

Flush all active transactions associated with the GSLB service when its state transitions from UP to DOWN. Do not enable this option for services that must complete their transactions. Applicable if connection proxy based site persistence is used.

Possible values: ENABLED, DISABLED

maxAAAUsers

Maximum number of SSL VPN users that can be logged on concurrently to the VPN virtual server that is

represented by this GSLB service. A GSLB service whose user count reaches the maximum is not considered when a GSLB decision is made, until the count drops below the maximum.

Maximum value: 65535

monThreshold

Monitoring threshold value for the GSLB service. If the sum of the weights of the monitors that are bound to this GSLB service and are in the UP state is not equal to or greater than this threshold value, the service is marked as DOWN.

Maximum value: 65535

hashId

Unique hash identifier for the GSLB service, used by hash based load balancing methods.

Minimum value: 1

comment

Any comments that you might want to associate with the GSLB service.

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

```
add gslb service sj_svc 203.12.123.12 http 80 -site san_jos
```

Removes a global server load balancing (GSLB) service configured on the appliance.

```
rm gslb service <serviceName>
```

serviceName

Name of the GSLB service.

```
rm gslb service sj_svc
```

Modifies the specified parameters of a global server load balancing (GSLB) service.

set gslb service <serviceName> [-IPAddress <ip_addr|ipv6_addr|*>] [-publicIP <ip_addr|ipv6_addr|*>] [-publicPort <port>] [-cip (ENABLED | DISABLED) [<cipHeader>]] [-sitePersistence <sitePersistence>] [-sitePrefix <string>] [-maxClient <positive_integer>] [-healthMonitor (YES | NO)] [-maxBandwidth <positive_integer>] [-downStateFlush (ENABLED | DISABLED)] [-maxAAAUsers <positive_integer>] [-viewName <string> <viewIP>] [-monThreshold <positive_integer>] [-weight <positive_integer> <monitorName>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)]

serviceName

Name of the GSLB service.

IPAddress

The new IP address of the service.

publicIP

The public IP address that a NAT device translates to the GSLB service's private IP address. Optional.

publicPort

The public port associated with the GSLB service's public IP address. The port is mapped to the service's private port number. Applicable to the local GSLB service. Optional.

Minimum value: 1

cip

In the request that is forwarded to the GSLB service, insert a header that stores the client's IP address. Client IP header insertion is used in connection-proxy based site persistence.

Possible values: ENABLED, DISABLED

Default value: DISABLED

sitePersistence

Use cookie-based site persistence. Applicable only to HTTP and SSL GSLB services.

Possible values: ConnectionProxy, HTTPRedirect, NONE

sitePrefix

The site's prefix string. When the service is bound to a GSLB virtual server, a GSLB site domain is generated internally for each bound service-domain pair by concatenating the site prefix of the service and the name of the domain. If the special string NONE is specified, the site-prefix string is unset. When implementing HTTP redirect site

persistence, the NetScaler appliance redirects GSLB requests to GSLB services by using their site domains.

maxClient

The maximum number of open connections that the service can support at any given time. A GSLB service whose connection count reaches the maximum is not considered when a GSLB decision is made, until the connection count drops below the maximum.

Maximum value: 4294967294

healthMonitor

Monitor the health of the GSLB service.

Possible values: YES, NO

Default value: YES

maxBandwidth

Maximum bandwidth.

downStateFlush

Flush all active transactions associated with the GSLB service when its state transitions from UP to DOWN. Do not enable this option for services that must complete their transactions. Applicable if connection proxy based site persistence is used.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxAAAUsers

Maximum number of SSL VPN users that can be logged on concurrently to the VPN virtual server that is represented by this GSLB service. A GSLB service whose user count reaches the maximum is not considered when a GSLB decision is made, until the count drops below the maximum.

Maximum value: 65535

viewName

Name of the DNS view of the service. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

monThreshold

Monitoring threshold value for the GSLB service. If the sum of the weights of the monitors that are bound to this GSLB service and are in the UP state is not equal to or greater than this threshold value, the service is marked as DOWN.

Maximum value: 65535

weight

Weight to assign to the monitor-service binding. A larger number specifies a greater weight. Contributes to the monitoring threshold, which determines the state of the service.

Minimum value: 1

Maximum value: 100

hashId

Unique hash identifier for the GSLB service, used by hash based load balancing methods.

Minimum value: 1

comment

Any comments that you might want to associate with the GSLB service.

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

```
set gslb service sj_svc  -sitePersistence ConnectionProxy
```

Use this command to remove gslb service settings.Refer to the set gslb service command for meanings of the arguments.

```
unset gslb service <serviceName> [-publicIP] [-publicPort] [-cip] [-cipHeader] [-sitePersistence] [-sitePrefix] [-maxClient] [-healthMonitor] [-maxBandwidth] [-downStateFlush] [-maxAAAUsers] [-viewIP] [-monThreshold] [-monitorName] [-hashId] [-comment] [-appflowLog]
```

Binds a DNS view or a monitor to a global server load balancing (GSLB) service.

```
bind gslb service <serviceName> ((-viewName <string> <viewIP>) | (-monitorName <string>@ [-monState ( ENABLED | DISABLED )] [-weight <positive_integer>]))
```

serviceName

Name of the GSLB service.

viewName

Name of the DNS view of the service. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

monitorName

Name of the monitor to bind to the GSLB service.

bind gslb service -viewName privateview 1.2.3.4

Unbinds a DNS view or a monitor from a global server load balancing (GSLB) service.

unbind gslb service <serviceName> (-viewName <string> | -monitorName <string>@)

serviceName

Name of the GSLB service.

viewName

Name of the DNS view of the service. A DNS view specifies the IP address that must be returned to clients accessing the service from a specific location.

monitorName

Name of the monitor to unbind.

unbind gslb service -viewName privateview

Displays the parameters of all the global server load balancing (GSLB) services configured on the appliance, or the parameters of just the specified service, and statistics related to the service. To display the parameters of all the GSLB services, do not specify a service name.

show gslb service [<serviceName>] show gslb service stats - alias for 'stat gslb service'

serviceName

Name of the GSLB service.

summary

fullValues

format

level

gslb

IPAddress

IP address of the service

serverName

Name of the server hosting the GSLB service.

serviceType

Service type.

port

Port number of the service.

publicIP

Public ip of the service

publicPort

Public port of the service

maxClient

Maximum number of clients.

maxAAAUsers

Maximum number of SSL VPN users that can be logged on concurrently to the VPN virtual server that is represented by this GSLB service. A GSLB service whose user count reaches the maximum is not considered when a GSLB decision is made, until the count drops below the maximum.

siteName

Name of the site to which the service belongs.

svrState

Server state.

svrEffGslbState

Effective state of the gslb svc

gslbthreshold

Indicates if gslb svc has reached threshold

gslbsvcStats

Indicates if gslb svc has stats of the primary or the whole chain

state

Enable or disable the service.

monitorName

Monitor name.

monState

The running state of the monitor on this service.

cip

Indicates if Client IP option is enabled

cipHeader

The client IP header used in the HTTP request.

sitePersistence

Indicates the type of cookie persistence set

sitePrefix

The site prefix string.

cltTimeout

Client timeout in seconds.

svrTimeout

Server timeout in seconds.

totalfailedprobes

The total number of failed probs.

preferredLocation

Prefered location.

maxBandwidth

Maximum bandwidth.

downStateFlush

Flush all active transactions associated with the GSLB service when its state transitions from UP to DOWN. Do not enable this option for services that must complete their transactions. Applicable if connection proxy based site persistence is used.

cnameEntry

Canonical name of the GSLB service. Used in CNAME-based GSLB.

viewName

Name of the DNS view of the service. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

viewIP

IP address to be used for the given view

weight

The Weight of monitor

monThreshold

Monitoring threshold value for the GSLB service. If the sum of the weights of the monitors that are bound to this GSLB service and are in the UP state is not equal to or greater than this threshold value, the service is marked as DOWN.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

responseTime

Response time of this monitor.

hashId

Unique hash identifier for the GSLB service, used by hash based load balancing methods.

comment

Any comments that you might want to associate with the GSLB service.

stateflag

stateflag

healthMonitor

Monitor the health of the GSLB service.

appflowLog

Enable logging appflow flow information

stateChangeTimeSec

Time when last state change happened. Seconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

monitorTotalProbes

Total number of probes sent to monitor this service.

monitorTotalFailedProbes

Total number of failed probes

monitorCurrentFailedProbes

Total number of currently failed probes

threshold

devno

count

show gslb service sj_svc

Displays the statistical data collected for a global server load balancing (GSLB) service.

stat gslb service [<serviceName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

serviceName

Name of the GSLB service.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Current Client Est connections (ClntEstConn)

Number of client connections in ESTABLISHED state.

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Service type (Type)

The service type of this service.Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP,

ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Current load on the service (Load)

Load on the service that is calculated from the bound load based monitor.

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Current client connections (ClntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Service hits (Hits)

Number of times that the service has been provided.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Renames a global server load balancing (GSLB) service.

rename gslb service <serviceName>@ <newName>@

serviceName

Existing name of the GSLB service.

newName

New name for the GSLB service.

rename gslb service gsl_svc gslb_svc_new

---

gslb site

Sep 22, 2015

The following operations can be performed on "gslb site":

add | rm | set | unset | show | stat

Creates a global server load balancing site.

add gslb site <siteName> [<siteType>] <siteIPAddress> [-publicIP <ip_addr|ipv6_addr|*>] [-metricExchange (ENABLED | DISABLED)] [-nwMetricExchange (ENABLED | DISABLED)] [-sessionExchange (ENABLED | DISABLED)] [-triggerMonitor <triggerMonitor>] [-parentSite <string>]

siteName

Name for the GSLB site. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my gslbsite" or 'my gslbsite').

siteType

Type of site to create. If the type is not specified, the appliance automatically detects and sets the type on the basis of the IP address being assigned to the site. If the specified site IP address is owned by the appliance (for example, a MIP address or SNIP address), the site is a local site. Otherwise, it is a remote site.

Possible values: REMOTE, LOCAL

Default value: NS_NORMAL

siteIPAddress

IP address for the GSLB site. The GSLB site uses this IP address to communicate with other GSLB sites. For a local site, use any IP address that is owned by the appliance (for example, a SNIP or MIP address, or the IP address of the ADNS service).

publicIP

Public IP address for the local site. Required only if the appliance is deployed in a private address space and the site has a public IP address hosted on an external firewall or a NAT device.

metricExchange

Exchange metrics with other sites. Metrics are exchanged by using Metric Exchange Protocol (MEP). The appliances

in the GSLB setup exchange health information once every second.

If you disable metrics exchange, you can use only static load balancing methods (such as round robin, static proximity, or the hash-based methods), and if you disable metrics exchange when a dynamic load balancing method (such as least connection) is in operation, the appliance falls back to round robin. Also, if you disable metrics exchange, you must use a monitor to determine the state of GSLB services. Otherwise, the service is marked as DOWN.

Possible values: ENABLED, DISABLED

Default value: ENABLED

nwMetricExchange

Exchange, with other GSLB sites, network metrics such as round-trip time (RTT), learned from communications with various local DNS (LDNS) servers used by clients. RTT information is used in the dynamic RTT load balancing method, and is exchanged every 5 seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessionExchange

Exchange persistent session entries with other GSLB sites every five seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

triggerMonitor

Specify the conditions under which the GSLB service must be monitored by a monitor, if one is bound. Available settings function as follows:

* ALWAYS - Monitor the GSLB service at all times.

* MEPDOWN - Monitor the GSLB service only when the exchange of metrics through the Metrics Exchange Protocol (MEP) is disabled.

MEPDOWN_SVCDOWN - Monitor the service in either of the following situations:

* The exchange of metrics through MEP is disabled.

* The exchange of metrics through MEP is enabled but the status of the service, learned through metrics exchange, is DOWN.

Possible values: ALWAYS, MEPDOWN, MEPDOWN_SVCDOWN

Default value: NSGSLB_TRIGMON_ALWAYS

parentSite

Parent site of the GSLB site, in a parent-child topology.


```
add site new_york  LOCAL  192.168.100.12 -publicIP  65.200.211.139
```

Removes a global server load balancing (GSLB) site and all its constituent GSLB services.

```
rm gslb site <siteName>
```

siteName

Name of the GSLB site to remove.

```
rm gslb site new_york
```

Modifies the specified parameters of a global server load balancing (GSLB) site.

```
set gslb site <siteName> [-metricExchange ( ENABLED | DISABLED )] [-nwMetricExchange ( ENABLED | DISABLED )] [-sessionExchange ( ENABLED | DISABLED )] [-triggerMonitor <triggerMonitor>]
```

siteName

Name of the GSLB site.

metricExchange

Exchange metrics with other sites. Metrics are exchanged by using Metric Exchange Protocol (MEP). The appliances in the GSLB setup exchange health information once every second.

If you disable metrics exchange, you can use only static load balancing methods (such as round robin, static proximity, or the hash-based methods), and if you disable metrics exchange when a dynamic load balancing method (such as least connection) is in operation, the appliance falls back to round robin. Also, if you disable metrics exchange, you must use a monitor to determine the state of GSLB services. Otherwise, the service is marked as DOWN.

Possible values: ENABLED, DISABLED

Default value: ENABLED

nwMetricExchange

Exchange, with other GSLB sites, network metrics such as round-trip time (RTT), learned from communications with various local DNS (LDNS) servers used by clients. RTT information is used in the dynamic RTT load balancing method, and is exchanged every 5 seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessionExchange

Exchange persistent session entries with other GSLB sites every five seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

triggerMonitor

Specify the conditions under which the GSLB service must be monitored by a monitor, if one is bound. Available settings function as follows:

* ALWAYS - Monitor the GSLB service at all times.

* MEPDOWN - Monitor the GSLB service only when the exchange of metrics through the Metrics Exchange Protocol (MEP) is disabled.

MEPDOWN_SVCDOWN - Monitor the service in either of the following situations:

* The exchange of metrics through MEP is disabled.

* The exchange of metrics through MEP is enabled but the status of the service, learned through metrics exchange, is DOWN.

Possible values: ALWAYS, MEPDOWN, MEPDOWN_SVCDOWN

Default value: NSGSLB_TRIGMON_ALWAYS

set gslb site new_york - metricExchange DISABLED

Use this command to remove gslb site settings.Refer to the set gslb site command for meanings of the arguments.

unset gslb site <siteName> [-metricExchange] [-nwMetricExchange] [-sessionExchange] [-triggerMonitor]

Displays the parameters of all the GSLB sites configured on the appliance, or the parameters of the specified GSLB site.

show gslb site [<siteName>] show gslb site stats - alias for 'stat gslb site'

siteName

Name of the GSLB site. If you specify a site name, details of all the site's constituent services are also displayed.

summary

fullValues

format

level

siteType

Specifies whether the site is LOCAL or REMOTE.

siteIPAddress

The IP address of the site.

publicIP

The Public IP of the gslb site.

metricExchange

Exchange metrics with other sites. Metrics are exchanged by using Metric Exchange Protocol (MEP). The appliances in the GSLB setup exchange health information once every second.

If you disable metrics exchange, you can use only static load balancing methods (such as round robin, static proximity, or the hash-based methods), and if you disable metrics exchange when a dynamic load balancing method (such as least connection) is in operation, the appliance falls back to round robin. Also, if you disable metrics exchange, you must use a monitor to determine the state of GSLB services. Otherwise, the service is marked as DOWN.

serviceName

Service name.

IPAddress

IP Address of the gslb service.

port

Port number of the gslb service.

state

State of the gslb service.

status

Current metric exchange status.

persistenceMEPStatus

Network metric and persistence exchange MEP connection status

serviceType

Service type.

nwMetricExchange

Specifies whether the exchange of network metrics like RTT is enabled or disabled.

sessionExchange

Specifies whether the exchange of persistence session entries is enabled or disabled.

triggerMonitor

Specify the conditions under which the GSLB service must be monitored by a monitor, if one is bound. Available settings function as follows:

* ALWAYS - Monitor the GSLB service at all times.

* MEPDOWN - Monitor the GSLB service only when the exchange of metrics through the Metrics Exchange Protocol (MEP) is disabled.

MEPDOWN_SVCDOWN - Monitor the service in either of the following situations:

* The exchange of metrics through MEP is disabled.

* The exchange of metrics through MEP is enabled but the status of the service, learned through metrics exchange, is DOWN.

parentSite

Parent site of the GSLB site, in a parent-child topology.

cnameEntry

The cname of the gslb service.

stateflag

stateflag

version

will be true if the remote site's version is ncore compatible with the local site.(>= 9.2)

devno

count

show site new_york

Displays statistics for a GSLB site.

stat gslb site [<siteName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

siteName

Name of the GSLB site for which to display detailed statistics. If a name is not specified, basic information about all GSLB sites is displayed.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Gslb Site Public IP address (Public IP)

The public IP address of this GSLB site.

Gslb Site private IP address (Private IP)

The private IP address of this GSLB site.

Metric Exchange State (MEPstate)

Indicates the status of the Metric Exchange Policy at this GSLB site.

Persistence Exchange (PersMEP)

Indicates whether Persistence entries exchange is enabled or disabled at this GSLB site.

Network Metric Exchange (NwMEP)

Indicates whether network metric exchange is enabled or disabled at this GSLB site.

Metric Exchange (MEP)

Indicates whether metric exchange is enabled or disabled at this GSLB site.

GSLB Site type (sitetype)

Indicates whether this GSLB site is local or remote.

Gslb Site public IP address (Public IP)

The public IP address of this GSLB site.

Site Metric Metric Exchange State (SiteMetricMEPstate)

Indicates the status of the site metric Metric Exchange connection at this GSLB site.

Netowrk Metric Metric Exchange State (NwMetricMEPstate)

Indicates the status of the network metric Metric Exchange connection at this GSLB site.

Request bytes (Reqb)

Total number of request bytes received by the virtual servers represented by all GSLB services associated with this GSLB site.

Response bytes (Rspb)

Number of response bytes received by the virtual servers represented by all GSLB services associated with this GSLB site.

Requests (Req)

Total number of requests received by the virtual servers represented by all GSLB services associated with this GSLB site.

Responses (Rsp)

Number of responses received by the virtual servers represented by all GSLB services associated with this GSLB site.

Current client connections (ClntConn)

Number of current client connections to the virtual servers represented by all GSLB services associated with this GSLB site.

Current server connections (SvrConn)

Number of current connections to the real servers behind the virtual servers represented by all GSLB services associated with this GSLB site.

gslb syncStatus

Sep 22, 2015

The following operations can be performed on "gslb syncStatus":

Displays the status of the last GSLB configuration synchronization.

show gslb syncStatus

response

gslb sync status as text blob

gslb vserver

Sep 22, 2015

The following operations can be performed on "gslb vserver":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

Creates a global server load balancing (GSLB) virtual server.

add gslb vserver <name> <serviceType> [-dnsRecordType <dnsRecordType>] [-lbMethod <lbMethod>] [-backupLBMethod <backupLBMethod>] [-netmask <netmask>] [-v6netmasklen <positive_integer>] [-tolerance <positive_integer>] [-persistenceType (SOURCEIP | NONE)] [-persistenceId <positive_integer>] [-persistMask <netmask>] [-v6persistmasklen <positive_integer>] [-timeout <mins>] [-EDR (ENABLED | DISABLED)] [-MIR (ENABLED | DISABLED)] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-dynamicWeight <dynamicWeight>] [-state (ENABLED | DISABLED)] [-considerEffectiveState (NONE | STATE_ONLY)] [-comment <string>] [-soMethod <soMethod>] [-soPersistence (ENABLED | DISABLED)] [-soPersistenceTimeOut <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-appflowLog (ENABLED | DISABLED)]

name

Name for the GSLB virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceType

Protocol used by services bound to the virtual server.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, NNTP, ANY, SIP_UDP, RADIUS, RDP, RTSP, MYSQL, MSSQL

ipType

The IP type for this GSLB vserver.

Possible values: IPV4, IPV6

Default value: NSGSLB_IPV4

dnsRecordType

DNS record type to associate with the GSLB virtual server's domain name.

Possible values: A, AAAA, CNAME

Default value: NSGSLB_A

lbMethod

Load balancing method for the GSLB virtual server.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

Default value: PEMGMT_LB_LEASTCONNS

backupSessionTimeout

A non zero value enables the feature whose minimum value is 2 minutes. The feature can be disabled by setting the value to zero. The created session is in effect for a specific client per domain.

Maximum value: 1440

backupLBMethod

Backup load balancing method. Becomes operational if the primary load balancing method fails or cannot be used. Valid only if the primary method is based on either round-trip time (RTT) or static proximity.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

netmask

IPv4 network mask for use in the SOURCEIPHASH load balancing method.

Default value: 0xFFFFFFFF

v6netmasklen

Number of bits to consider, in an IPv6 source IP address, for creating the hash that is required by the SOURCEIPHASH load balancing method.

Default value: 128

Minimum value: 1

Maximum value: 128

tolerance

Site selection tolerance, in milliseconds, for implementing the RTT load balancing method. If a site's RTT deviates from the lowest RTT by more than the specified tolerance, the site is not considered when the NetScaler appliance makes a GSLB decision. The appliance implements the round robin method of global server load balancing between sites whose RTT values are within the specified tolerance. If the tolerance is 0 (zero), the appliance always sends clients the IP address of the site with the lowest RTT.

Maximum value: 100

persistenceType

Use source IP address based persistence for the virtual server.

After the load balancing method selects a service for the first packet, the IP address received in response to the DNS query is used for subsequent requests from the same client.

Possible values: SOURCEIP, NONE

persistenceId

The persistence ID for the GSLB virtual server. The ID is a positive integer that enables GSLB sites to identify the GSLB virtual server, and is required if source IP address based or spill over based persistence is enabled on the virtual server.

Maximum value: 65535

persistMask

The optional IPv4 network mask applied to IPv4 addresses to establish source IP address based persistence.

Default value: 0xFFFFFFFF

v6persistmasklen

Number of bits to consider in an IPv6 source IP address when creating source IP address based persistence sessions.

Default value: 128

Minimum value: 1

Maximum value: 128

timeout

Idle time, in minutes, after which a persistence entry is cleared.

Default value: 2

Minimum value: 2

Maximum value: 1440

EDR

Send clients an empty DNS response when the GSLB virtual server is DOWN.

Possible values: ENABLED, DISABLED

Default value: DISABLED

MIR

Include multiple IP addresses in the DNS responses sent to clients.

Possible values: ENABLED, DISABLED

Default value: DISABLED

disablePrimaryOnDown

Continue to direct traffic to the backup chain even after the primary GSLB virtual server returns to the UP state. Used when spillover is configured for the virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicWeight

Specify if the appliance should consider the service count, service weights, or ignore both when using weight-based load balancing methods. The state of the number of services bound to the virtual server help the appliance to select the service.

Possible values: SERVICECOUNT, SERVICEWEIGHT, DISABLED

Default value: DISABLED

state

State of the GSLB virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

considerEffectiveState

If the primary state of all bound GSLB services is DOWN, consider the effective states of all the GSLB services, obtained through the Metrics Exchange Protocol (MEP), when determining the state of the GSLB virtual server. To consider the effective state, set the parameter to STATE_ONLY. To disregard the effective state, set the parameter to NONE.

The effective state of a GSLB service is the ability of the corresponding virtual server to serve traffic. The effective state of the load balancing virtual server, which is transferred to the GSLB service, is UP even if only one virtual server in the backup chain of virtual servers is in the UP state.

Possible values: NONE, STATE_ONLY

Default value: NS_GSLB_DONOT_CONSIDER_BKPS

comment

Any comments that you might want to associate with the GSLB virtual server.

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the GSLB virtual server exceeds the sum of the maximum client (Max Clients) settings for bound GSLB services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of the bound GSLB services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the GSLB virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the GSLB services that are UP drops below the threshold. For example, if services gslbSvc1, gslbSvc2, and gslbSvc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if gslbSvc1 and gslbSvc3 or gslbSvc2 and gslbSvc3 transition to DOWN.

* NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup GSLB virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

```
add gslb vserver gvip http
```

Removes a global server load balancing (GSLB) virtual server configured on the appliance.

rm gslb vserver <name>

name

Name of the GSLB virtual server to remove.

rm gslb vserver gvip

Modifies the specified parameters of a global server load balancing (GSLB) virtual server.

set gslb vserver <name> [-dnsRecordType <dnsRecordType>] [-backupVServer <string>] [-lbMethod <lbMethod>] [-backupLBMethod <backupLBMethod>] [-netmask <netmask>] [-v6netmasklen <positive_integer>] [-tolerance <positive_integer>] [-persistenceType (SOURCEIP | NONE)] [-persistenceId <positive_integer>] [-persistMask <netmask>] [-v6persistmasklen <positive_integer>] [-timeout <mins>] [-EDR (ENABLED | DISABLED)] [-MIR (ENABLED | DISABLED)] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-dynamicWeight <dynamicWeight>] [-considerEffectiveState (NONE | STATE_ONLY)] [-soMethod <soMethod>] [-soPersistence (ENABLED | DISABLED)] [-soPersistenceTimeOut <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-serviceName <string> -weight <positive_integer>] [-domainName <string> [-TTL <secs>] [-backupIP <ip_addr|ipv6_addr|*>] [-cookieDomain <string>] [-cookieTimeout <mins>] [-sitedomainTTL <secs>]] [-comment <string>] [-appflowLog (ENABLED | DISABLED)]

name

Name of the GSLB virtual server.

ipType

The IP type for this GSLB vserver.

Possible values: IPV4, IPV6

Default value: NSGSLB_IPV4

dnsRecordType

DNS record type to associate with the GSLB virtual server's domain name.

Possible values: A, AAAA, CNAME

Default value: NSGSLB_A

backupVServer

Name of the backup GSLB virtual server to which the appliance should to forward requests if the status of the primary GSLB virtual server is down or exceeds its spillover threshold.

backupSessionTimeout

A non zero value enables the feature whose minimum value is 2 minutes. The feature can be disabled by setting the value to zero. The created session is in effect for a specific client per domain.

Maximum value: 1440

lbMethod

Load balancing method for the GSLB virtual server.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

Default value: PEMGMT_LB_LEASTCONNS

netmask

IPv4 network mask for use in the SOURCEIPHASH load balancing method.

Default value: 0xFFFFFFFF

v6netmasklen

Number of bits to consider, in an IPv6 source IP address, for creating the hash that is required by the SOURCEIPHASH load balancing method.

Default value: 128

Minimum value: 1

Maximum value: 128

tolerance

Site selection tolerance, in milliseconds, for implementing the RTT load balancing method. If a site's RTT deviates from the lowest RTT by more than the specified tolerance, the site is not considered when the NetScaler appliance makes a GSLB decision. The appliance implements the round robin method of global server load balancing between sites whose RTT values are within the specified tolerance. If the tolerance is 0 (zero), the appliance always sends clients the IP address of the site with the lowest RTT.

Maximum value: 100

persistenceType

Persistence type for the virtual server. Possible value for this parameter is SOURCEIP, which specifies persistence based on the source IP address of inbound packets. After the load balancing method selects a link for transmission of the first packet, the IP address received in response to the DNS query is used for subsequent requests from the same client.

Possible values: SOURCEIP, NONE

persistenceId

The persistence ID for the GSLB virtual server. The ID is a positive integer that enables GSLB sites to identify the GSLB virtual server, and is required if source IP address based or spill over based persistence is enabled on the virtual server.

Maximum value: 65535

persistMask

The optional IPv4 network mask applied to IPv4 addresses to establish source IP address based persistence.

Default value: 0xFFFFFFFF

v6persistmasklen

Number of bits to consider in an IPv6 source IP address when creating source IP address based persistence sessions.

Default value: 128

Minimum value: 1

Maximum value: 128

timeout

Idle time, in minutes, after which a persistence entry is cleared.

Default value: 2

Minimum value: 2

Maximum value: 1440

EDR

Send clients an empty DNS response when the GSLB virtual server is DOWN.

Possible values: ENABLED, DISABLED

Default value: DISABLED

MIR

Include multiple IP addresses in the DNS responses sent to clients.

Possible values: ENABLED, DISABLED

Default value: DISABLED

disablePrimaryOnDown

Continue to direct traffic to the backup chain even after the primary GSLB virtual server returns to the UP state. Used when spillover is configured for the virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicWeight

Specify if the appliance should consider the service count, service weights, or ignore both when using weight-based load balancing methods. The state of the number of services bound to the virtual server help the appliance to select the service.

Possible values: SERVICECOUNT, SERVICEWEIGHT, DISABLED

Default value: DISABLED

considerEffectiveState

If the primary state of all bound GSLB services is DOWN, consider the effective states of all the GSLB services, obtained through the Metrics Exchange Protocol (MEP), when determining the state of the GSLB virtual server. To consider the effective state, set the parameter to STATE_ONLY. To disregard the effective state, set the parameter to NONE.

The effective state of a GSLB service is the ability of the corresponding virtual server to serve traffic. The effective state of the load balancing virtual server, which is transferred to the GSLB service, is UP even if only one virtual server in the backup chain of virtual servers is in the UP state.

Possible values: NONE, STATE_ONLY

Default value: NS_GSLB_DONOT_CONSIDER_BKPS

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the GSLB virtual server exceeds the sum of the maximum client (Max Clients) settings for bound GSLB services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of the bound GSLB services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the GSLB virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the GSLB services that are UP drops below the threshold. For example, if services gslbSvc1, gslbSvc2, and gslbSvc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if gslbSvc1 and gslbSvc3 or gslbSvc2 and gslbSvc3 transition to DOWN.

* NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup GSLB virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

serviceName

Name of the GSLB service for which to change the weight.

domainName

Domain name for which to change the time to live (TTL) and/or backup service IP address.

comment

Any comments that you might want to associate with the GSLB virtual server.

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

```
set gslb vserver gvip -persistenceType SOURCEIP
```

Removes the specified settings from the specified global server load balancing (GSLB) virtual server. Attributes for which a default value is available revert to their default values..Refer to the set gslb vserver command for meanings of the arguments.

unset gslb vserver <name>@ [-backupVServer] [-dnsRecordType] [-lbMethod] [-backupLBMethod] [-netmask] [-v6netmasklen] [-tolerance] [-persistenceType] [-persistenceId] [-persistMask] [-v6persistmasklen] [-timeout] [-EDR] [-MIR] [-disablePrimaryOnDown] [-dynamicWeight] [-considerEffectiveState] [-soMethod] [-soPersistence] [-soPersistenceTimeOut] [-soBackupAction] [-serviceName] [-weight] [-comment] [-appflowLog]

unset gslb vserver lb_vip -backupVServer For multiple gslb vservers the command is: unset gslb vserver lb_vip[1-3] -backupVServer

---

Binds a domain, service, backup IP address, or cookie domain to a GSLB virtual server.

bind gslb vserver <name> ((-serviceName <string> [-weight <positive_integer>]) | (-domainName <string> [-TTL <secs>] [-backupIP <ip_addr|ipv6_addr|*>] [-cookieDomain <string>] [-cookieTimeout <mins>] [-sitedomainTTL <secs>]) | (-policyName <string>@ [-priority <positive_integer>])) [-gotoPriorityExpression <expression>]

name

Name of the virtual server on which to perform the binding operation.

serviceName

Name of the GSLB service for which to change the weight.

domainName

Domain name for which to change the time to live (TTL) and/or backup service IP address.

policyName

Name of the policy bound to the GSLB vserver.

bind gslb vserver gvip -domainName www.mynw.com

Unbinds the domain or service from the GSLB virtual server.

unbind gslb vserver <name> (-serviceName <string> | (-domainName <string> [-backupIP] [-cookieDomain]) | -policyName <string>@)

name

Name of the GSLB virtual server.

serviceName

Name of the GSLB service for which to change the weight.

domainName

Domain name for which to change the time to live (TTL) and/or backup service IP address.

policyName

The policy that has been bound to this load balancing virtual server, using the ###bind gslb vserver### command.

unbind gslb vserver gvip -domainName www.mynw.com

Enables a global server load balancing (GSLB) virtual server that has been disabled. (A GSLB virtual server is enabled by default.)

enable gslb vserver <name>@

name

Name of the GSLB virtual server to enable.

enable gslb vserver gslb_vip To enable multiple gslb vservers use the following command: enable gslb vserver gslb_vip[1-3]

Disables a global server load balancing (GSLB) virtual server and takes it out of service.

disable gslb vserver <name>@

name

Name of the GSLB virtual server to disable.

disable gslb vserver gslb_vip To disable multiple gslb vservers use the following command: disable gslb vserver gslb_vip[1-3]

Displays the parameters of all the global server load balancing (GSLB) virtual servers configured on the appliance, or the parameters of the specified GSLB virtual server.

show gslb vserver [<name>] show gslb vserver stats - alias for 'stat gslb vserver'

name

Name of the GSLB virtual server.

summary

fullValues

format

level

serviceType

Protocol used by services bound to the virtual server.

ipType

The IP type for this GSLB vserver.NOTE: This attribute is deprecated.

dnsRecordType

The IP type for this GSLB vserver.

persistenceType

Indicates if persistence is set on the gslb vserver

persistenceId

Persistence id of the gslb vserver

lbMethod

The load balancing method set for the virtual server

backupLBMethod

Indicates the backup method in case the primary fails

tolerance

Indicates the deviation we can tolerate when we have the LB method as RTT

timeout

Idle timeout for persistence entries.

state

State of the gslb vserver.

netmask

The netmask used in the SOURCEIPHASH policy.

v6netmasklen

The netmask used for ipv6 traffic in the SOURCE/DEST IPHASH policy.

persistMask

The netmask used while SOURCEIP based persistency is ENABLED.

v6persistmasklen

The netmask applied for ipv6 traffic when the persistency type is SOURCEIP.

serviceName

The service name.

weight

Weight for the service.

domainName

The name of the domain for which TTL and/or backupIP has changed.

TTL

TTL for the given domain.

backupIP

Backup IP for the given domain.

cookieDomain

cookieTimeout

Time out value of the cookie in minutes

sitedomainTTL

Site domain TTL.

IPAddress

IP address.

port

Port number.

status

Current status of the gslb vserver. During the initial phase if the configured lb method is not round robin , the vserver will adopt round robin to distribute traffic for a predefined number of requests.

lbrrreason

Reason why a vserver is in RR. The following are the reasons:

1 - MEP is DOWN (GSLB)

2 - LB method has changed

3 - Bound service's state changed to UP

4 - A new service is bound

5 - Startup RR factor has changed

6 - LB feature is enabled

7 - Load monitor is not active on a service

8 - Vserver is Enabled

9 - SSL feature is Enabled

10 - All bound services have reached threshold. Using effective state to load balance (GSLB)

11 - Primary state of bound services are not UP. Using effective state to load balance (GSLB)

12 - No LB decision can be made as all bound services have either reached threshold or are not UP (GSLB)

13 - All load monitors are active

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard `*' is accepted as a valid qualifier token.

backupVServer

Backup vserver in case the primary fails

backupSessionTimeout

A non zero value enables the feature. The minimum value is 2 minutes. To disable the feature set the value to zero. The created session is in effect for a specific client per domain.NOTE: This attribute is deprecated.This is a deprecated attribute.

EDR

Indicates if Empty Down Response is enabled/disabled

MIR

Indicates if Multi IP Response is enabled/disabled

disablePrimaryOnDown

Continue to direct traffic to the backup chain even after the primary GSLB virtual server returns to the UP state. Used when spillover is configured for the virtual server.

dynamicWeight

Dynamic weight method. Possible values are, the svc count or the svc weights or ignore both.

isCname

is cname feature set on vserver

cumulativeWeight

Cumulative weight is the weight of GSLB service considering both its configured weight and dynamic weight. It is equal to product of dynamic weight and configured weight of the gslb service

dynamicConfWt

Weight obtained by the virtue of bound service count or weight

thresholdValue

Tells whether threshold exceeded for this service participating in CUSTOMLB

sitePersistence

Type of Site Persistence set

svrEffGslbState

Effective state of the gslb svc

gslbthreshold

Indicates if gslb svc has reached threshold

considerEffectiveState

If the primary state of all bound GSLB services is DOWN, consider the effective states of all the GSLB services, obtained through the Metrics Exchange Protocol (MEP), when determining the state of the GSLB virtual server. To consider the effective state, set the parameter to STATE_ONLY. To disregard the effective state, set the parameter to NONE.

The effective state of a GSLB service is the ability of the corresponding virtual server to serve traffic. The effective state of the load balancing virtual server, which is transferred to the GSLB service, is UP even if only one virtual server in the backup chain of virtual servers is in the UP state.

cnameEntry

The cname of the gslb service.

totalServices

Total number of services bound to the vserver.

activeServices

Total number of active services bound to the vserver.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimemSec

Time at which last state change happened. Milliseconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

comment

Any comments that you might want to associate with the GSLB virtual server.

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the GSLB virtual server exceeds the sum of the maximum client (Max Clients) settings for bound GSLB services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of the bound GSLB services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the GSLB virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the GSLB services that are UP drops below the threshold. For example, if services gslbSvc1, gslbSvc2, and gslbSvc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if gslbSvc1 and gslbSvc3 or gslbSvc2 and gslbSvc3 transition to DOWN.

* NONE - Spillover does not occur.

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup GSLB virtual servers.

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

health

Health of vserver based on percentage of weights of active svcs/all svcs. This does not consider administratively disabled svcs

stateflag

stateflag

appflowLog

Enable logging appflow flow information

policyName

Name of the policy bound to the GSLB vserver.

priority

Priority.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

o If gotoPriorityExpression is not present or if it is equal to END then the policy bank evaluation ends here

o Else if the gotoPriorityExpression is equal to NEXT then the next policy in the priority order is evaluated.

o Else gotoPriorityExpression is evaluated. The result of gotoPriorityExpression (which has to be a number) is processed as follows:

- An UNDEF event is triggered if

. gotoPriorityExpression cannot be evaluated

. gotoPriorityExpression evaluates to number which is smaller than the maximum priority in the policy bank but is not same as any policy's priority

. gotoPriorityExpression evaluates to a priority that is smaller than the current policy's priority

- If the gotoPriorityExpression evaluates to the priority of the current policy then the next policy in the priority order is evaluated.

- If the gotoPriorityExpression evaluates to the priority of a policy further ahead in the list then that policy will be evaluated next.

This field is applicable only to rewrite and responder policies.

type

The bindpoint to which the policy is bound

devno

count

show gslb vserver gvip

Displays statistics associated with a global server load balancing (GSLB) virtual server.

stat gslb vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the GSLB virtual server for which to display statistics. If you do not specify a name, statistics are displayed for all GSLB virtual servers.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Current Client Est connections (ClntEstConn)

Number of client connections in ESTABLISHED state.

total INACTIVE services (inactSvcs)

number of INACTIVE services bound to a vserver

Vserver Health (Health)

Health of the vserver. This gives percentage of UP services bound to this vserver.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

total ACTIVE services (actSvcs)

number of ACTIVE services bound to a vserver

Vserver hits (Hits)

Total vserver hits

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Spill Over Threshold (SOThresh)

Spill Over Threshold set on the VServer.

Spill Over Hits (NumSo)

Number of times vserver experienced spill over.

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Current client connections (ClntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Renames a global server load balancing (GSLB) virtual server.

rename gslb vserver <name>@ <newName>@

name

Existing name of the GSLB virtual server.

newName

New name for the GSLB virtual server.

```
rename gslb vserver gsl_vsvr gslb_vsvr_new
```

High Availability Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- HA failover
- HA files
- HA node
- HA sync

HA failover

Sep 22, 2015

The following operations can be performed on "HA failover":

Forces an HA failover. Can be initiated from either node. A forced failover is not propagated or synchronized., Note: This command fails under any of the following conditions: * The secondary node is disabled or configured to remain secondary. * The primary node is configured to remain primary. * The state of the peer node is unknown. * You run the command on a standalone appliance.

force HA failover [-force]

force

Force a failover without prompting for confirmation.

HA files

Sep 22, 2015

The following operations can be performed on "HA files":

Synchronize various configuration files from the primary node to the secondary. You can run this command from either node. Files that are present on only the secondary and are specific to the secondary are not deleted. This command fails if the secondary node is disabled, the secondary node is not accessible from the primary, or you enter the command on a standalone appliance.

sync HA files [<Mode> ...]

Mode

Specify one of the following modes of synchronization.

* all - Synchronize files related to system configuration, Access Gateway bookmarks, SSL certificates, SSL CRL lists, HTML injection scripts, and Application Firewall XML objects.

* bookmarks - Synchronize all Access Gateway bookmarks.

* ssl - Synchronize all certificates, keys, and CRLs for the SSL feature.

* htmlinjection. Synchronize all scripts configured for the HTML injection feature.

* imports. Synchronize all XML objects (for example, WSDLs, schemas, error pages) configured for the application firewall.

* misc - Synchronize all license files and the rc.conf file.

* all_plus_misc - Synchronize files related to system configuration, Access Gateway bookmarks, SSL certificates, SSL CRL lists, HTML injection scripts, application firewall XML objects, licenses, and the rc.conf file.

sync files all

HA node

Sep 22, 2015

The following operations can be performed on "HA node":

add | rm | set | unset | bind | unbind | show | stat

Adds a peer node to an HA configuration. Each node must add the other as a peer. An algorithm determines which node becomes primary and which becomes secondary.

add HA node <id> <IPAddress> [-inc (ENABLED | DISABLED)]

id

Number that uniquely identifies the node. For self node, it will always be 0. Peer node values can range from 1-64.

Minimum value: 1

Maximum value: 64

IPAddress

The NSIP or NSIP6 address of the node to be added for an HA configuration. This setting is neither propagated nor synchronized.

inc

This option is required if the HA nodes reside on different networks. When this mode is enabled, the following independent network entities and configurations are neither propagated nor synced to the other node: MIPs, SNIPs, VLANs, routes (except LLB routes), route monitors, RNAT rules (except any RNAT rule with a VIP as the NAT IP), and dynamic routing configurations. They are maintained independently on each node.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Removes the peer node from the HA configuration. To completely remove both the nodes from the HA configuration, you have to log on to each node and remove its peer node.

rm HA node <id>

id

Number that uniquely identifies the peer node.

CLI users: To learn the ID of the peer node, run the show HA node command on the local node.

Maximum value: 64

Sets the specified HA related parameters for the node. The settings are neither propagated nor synchronized to the peer node.

set HA node [-haStatus <haStatus>] [-haSync (ENABLED | DISABLED)] [-haProp (ENABLED | DISABLED)] [-helloInterval <msecs>] [-deadInterval <secs>] [-failSafe (ON | OFF)] [-maxFlips <positive_integer>] [-maxFlipTime <positive_integer>] [-syncvlan <positive_integer>]

id

Number that uniquely identifies the node. For self node, it will always be 0. Peer node values can range from 1-64.

Maximum value: 64

haStatus

The HA status of the node. The HA status STAYSECONDARY is used to force the secondary device stay as secondary independent of the state of the Primary device. For example, in an existing HA setup, the Primary node has to be upgraded and this process would take few seconds. During the upgradation, it is possible that the Primary node may suffer from a downtime for a few seconds. However, the Secondary should not take over as the Primary node. Thus, the Secondary node should remain as Secondary even if there is a failure in the Primary node.

STAYPRIMARY configuration keeps the node in primary state in case if it is healthy, even if the peer node was the primary node initially. If the node with STAYPRIMARY setting (and no peer node) is added to a primary node (which has this node as the peer) then this node takes over as the new primary and the older node becomes secondary. ENABLED state means normal HA operation without any constraints/preferences. DISABLED state disables the normal HA operation of the node.

Possible values: ENABLED, STAYSECONDARY, DISABLED, STAYPRIMARY

haSync

Automatically maintain synchronization by duplicating the configuration of the primary node on the secondary node. This setting is not propagated. Automatic synchronization requires that this setting be enabled (the default) on the current secondary node. Synchronization uses TCP port 3010.

Possible values: ENABLED, DISABLED

Default value: ENABLED

haProp

Automatically propagate all commands from the primary to the secondary node, except the following:

* All HA configuration related commands. For example, add ha node, set ha node, and bind ha node.

* All Interface related commands. For example, set interface and unset interface.

* All channels related commands. For example, add channel, set channel, and bind channel.

The propagated command is executed on the secondary node before it is executed on the primary. If command propagation fails, or if command execution fails on the secondary, the primary node executes the command and logs an error. Command propagation uses port 3010.

Note: After enabling propagation, run force synchronization on either node.

Possible values: ENABLED, DISABLED

Default value: ENABLED

helloInterval

Interval, in milliseconds, between heartbeat messages sent to the peer node. The heartbeat messages are UDP packets sent to port 3003 of the peer node.

Default value: 200

Minimum value: 200

Maximum value: 1000

deadInterval

Number of seconds after which a peer node is marked DOWN if heartbeat messages are not received from the peer node.

Default value: 3

Minimum value: 3

Maximum value: 60

failSafe

Keep one node primary if both nodes fail the health check, so that a partially available node can back up data and handle traffic. This mode is set independently on each node.

Possible values: ON, OFF

Default value: OFF

maxFlips

Max number of flips allowed before becoming sticky primary

maxFlipTime

Interval after which flipping of node states can again start

syncvlan

Vlan on which HA related communication is sent. This include sync, propagation , connection mirroring , LB persistency config sync, persistent session sync and session state sync. However HA heartbeats can go all interfaces.

Minimum value: 1

Maximum value: 4094

Use this command to remove HA node settings.Refer to the set HA node command for meanings of the arguments.

unset HA node [-haStatus] [-haSync] [-haProp] [-helloInterval] [-deadInterval] [-failSafe] [-maxFlips] [-maxFlipTime] [-syncvlan]

Adds a route monitor to the local node. When a NetScaler appliance has only static routes for reaching a network, and you want to create a route monitor for the network, you must enable monitored static routes (MSR) for the static routes. Route Monitors are supported both in non-INC and INC modes.

bind HA node [<id>] (-routeMonitor <ip_addr|ipv6_addr|*> [<netmask>])

id

Number that uniquely identifies the local node. The ID of the local node is always 0.

Maximum value: 64

routeMonitor

A route that you want the NetScaler appliance to monitor in its internal routing table. You can specify an IPv4 address or network, or an IPv6 address or network prefix. If you specify an IPv4 network address or IPv6 network prefix, the appliance monitors any route that matches the network or prefix.

Removes a route monitor entry from the local node. The NetScaler appliance stops monitoring the route in its internal routing table.

unbind HA node [<id>] (-routeMonitor <ip_addr|ipv6_addr|*> [<netmask>])

id

Number that uniquely identifies the local node. The ID of the local node is always 0.

Maximum value: 64

routeMonitor

The route specified in the route monitor entry that you want to remove from the NetScaler appliance. Can be an IPv4 address or network, or an IPv6 address or network prefix.

Displays the HA settings of both nodes or, if you specify a node, just the specified node. You can use this command to display the master state (primary or secondary) of the nodes in a HA configuration.

show HA node [<id>]

id

ID of the node whose HA settings you want to display. (The ID of the local node is always 0.)

Maximum value: 64

summary

fullValues

format

level

name

Node Name.

IPAddress

IP Address of the node.

flags

The flags for this entry.

stateflag

haStatus

HA status.

state

HA Master State.

haSync

HA Sync State.

haProp

HA Propagation Status.

enaifaces

Enabled interfaces.

disifaces

Disabled interfaces.

hamonifaces

HAMON ON interfaces.

pfifaces

Interfaces causing Partial Failure.

ifaces

Interfaces on which non-multicast is not seen.

network

The network.

netmask

The netmask.

inc

INC state.

ssl2

SSL card status.

helloInterval

Hello Interval.

deadInterval

Dead Interval.

masterStateTime

Time elapsed in current master state

failSafe

Keep one node primary if both nodes fail the health check, so that a partially available node can back up data and handle traffic. This mode is set independently on each node.

routeMonitor

The IP address (IPv4 or IPv6).

maxFlips

Max number of flips allowed before becoming sticky primary

maxFlipTime

Interval after which flipping of node states can again start

curFlips

Keeps track of number of flips that have happened till now in current interval

completedFlipTime

To inform user whether flip time is elapsed or not

syncvlan

Vlan on which HA related communication is sent. This include sync, propagation , connection mirroring , LB persistency config sync, persistent session sync and session state sync. However HA heartbeats can go all interfaces.

routeMonitorState

State for route monitor

devno

count

An example of the command's output is as follows: 2 configured nodes: 1) Node ID: 0 IP: 192.168.100.5 Primary node 2) Node ID: 2 IP

Display the statistics related to HA configuration.

stat HA node [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

High Availability (HA)

Whether a NetScaler appliance is configured for high availability. Possible values are YES and NO. If the value is NO, the high availability statistics below are invalid.

System state (HAState)

State of the HA node, based on its health, in a high availability setup. Possible values are:

UP ? Indicates that the node is accessible and can function as either a primary or secondary node.

DISABLED ? Indicates that the high availability status of the node has been manually disabled. Synchronization and propagation cannot take place between the peer nodes.

INIT ? Indicates that the node is in the process of becoming part of the high availability configuration.

PARTIALFAIL ? Indicates that one of the high availability monitored interfaces has failed because of a card or link failure. This state triggers a failover.

COMPLETEFAIL ? Indicates that all the interfaces of the node are unusable, because the interfaces on which high availability monitoring is enabled are not connected or are manually disabled. This state triggers a failover.

DUMB ? Indicates that the node is in listening mode. It does not participate in high availability transitions or transfer configuration from the peer node. This is a configured value, not a statistic.

PARTIALFAILSSL ? Indicates that the SSL card has failed. This state triggers a failover.

ROUTEMONITORFAIL ? Indicates that the route monitor has failed. This state triggers a failover.

Master state (mastate)

Indicates the high availability state of the node. Possible values are:

STAYSECONDARY ? Indicates that the selected node remains the secondary node in a high availability setup. In this case a forced failover does not change the state but, instead, returns an appropriate error message. This is a configured value and not a statistic.

PRIMARY ? Indicates that the selected node is the primary node in a high availability setup.

SECONDARY ? Indicates that the selected node is the secondary node in a high availability setup.

CLAIMING ? Indicates that the secondary node is in the process of taking over as the primary node. This is the intermediate state in the transition of the secondary node to primary status.

FORCE CHANGE - Indicates that the secondary node is forcibly changing its status to primary due to a forced failover issued on the secondary node.

Last Transition time (TransTime)

Time when the last master state transition occurred. You can use this statistic for debugging.

Heartbeats received (HApktrx)

Number of heartbeat packets received from the peer node. Heartbeats are sent at regular intervals (default is 200 milliseconds) to determine the state of the peer node.

Heartbeats sent (HApkttx)

Number of heartbeat packets sent to the peer node. Heartbeats are sent at regular intervals (default is 200 milliseconds) to determine the state of

the peer node.

Propagation timeouts (ptimeout)

Number of times propagation timed out.

Sync failure (syncfail)

Number of times the configuration of the primary and secondary nodes failed to synchronize since that last transition. A synchronization failure results in mismatched configuration. It can be caused by a mismatch in the Remote Procedural Call (RPC) password on the two nodes forming the high availability pair.

HA sync

Sep 22, 2015

The following operations can be performed on "HA sync":

Forces duplication of the primary node's configuration on the secondary node. Can be executed from either node. Note: This command fails under any of the following conditions: * Synchronization is already in progress. * The secondary node is disabled. * Synchronization is disabled on either node * The secondary node is not accessible from the primary. * You run the command on a standalone appliance.

force HA sync [-force [-save (YES | NO)]]

force

Force synchronization regardless of the state of HA propagation and HA synchronization on either node.

save

After synchronization, automatically save the configuration in the secondary node configuration file (ns.conf) without prompting for confirmation.

Possible values: YES, NO

Default value: VAL_NOT_SET

Can be used in following formats: >force sync <cr> >force sync -force <cr> >force sync -force -save [yes|no]<cr>

IPSec Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- ipsec counters
- ipsec parameter
- ipsec profile

ipsec counters

Sep 22, 2015

The following operations can be performed on "ipsec counters":

stat ipsec counters

Display statistics for secure tunnel sessions.

Synopsys

stat ipsec counters [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

Bytes Received (ipsecRxBytes)

Bytes received during IPsec sessions.

Bytes Sent (ipsecTxBytes)

Bytes sent during IPsec sessions.

Packets Received (ipsecRxPkts)

Packets received during IPsec sessions.

Packets Sent (ipsecTxPkts)

Packets sent during IPsec sessions.

Example

stat ipsec

ipsec parameter

Sep 22, 2015

The following operations can be performed on "ipsec parameter":

set | unset | show

set ipsec parameter

Set global parameters for IPSEC

Synopsys

set ipsec parameter [-ikeVersion (V1 | V2)] [-encAlgo (AES | 3DES) ...] [-hashAlgo <hashAlgo> ...] [-lifetime <positive_integer>] [-livenessCheckInterval <positive_integer>] [-replayWindowSize <positive_integer>] [-ikeRetryInterval <positive_integer>] [-retransmissiontime <positive_integer>]

Arguments

ikeVersion

IKE Protocol Version

Possible values: V1, V2

Default value: KMP_IKEV2

encAlgo

Type of encryption algorithm

Default value: ENC_AES

hashAlgo

Type of hashing algorithm

Default value: HMAC_SHA256

lifetime

Lifetime of SA in seconds

Minimum value: 60

Maximum value: 31536000

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveliness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveliness checks.

Maximum value: 64999

replayWindowSize

IPSec Replay window size for the data traffic

Maximum value: 16384

ikeRetryInterval

IKE retry interval for bringing up the connection

Minimum value: 60

Maximum value: 3600

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.,

increases for every retransmit till 6 retransmits.

Minimum value: 1

Maximum value: 99

---

unset ipsec parameter

Set global parameters for IPSEC.Refer to the set ipsec parameter command for meanings of the arguments.

Synopsys

unset ipsec parameter [-ikeVersion] [-encAlgo] [-hashAlgo] [-lifetime] [-livenessCheckInterval] [-replayWindowSize] [-ikeRetryInterval] [-retransmissiontime]

---

show ipsec parameter

Show global parameters for IPSEC

Synopsys

show ipsec parameter

Arguments

summary

fullValues

format

level

Outputs

ikeVersion

IKE Protocol Version

encAlgo

Type of encryption algorithm

hashAlgo

Type of hashing algorithm

lifetime

Lifetime of SA in seconds

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveliness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveliness checks.

replayWindowSize

IPSec Replay window size for the data traffic

ikeRetryInterval

IKE retry interval for bringing up the connection

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.,

increases for every retransmit till 6 retransmits.

ipsec profile

Sep 22, 2015

The following operations can be performed on "ipsec profile":

add | show | rm

add ipsec profile

Add an ipsec profile.

Synopsys

add ipsec profile <name> [-ikeVersion (V1 | V2)] [-encAlgo (AES | 3DES) ...] [-hashAlgo <hashAlgo> ...] [-lifetime <positive_integer>] (-psk | (-publickey <string> -privatekey <string> -peerPublicKey <string>)) [-livenessCheckInterval <positive_integer>] [-replayWindowSize <positive_integer>] [-ikeRetryInterval <positive_integer>] [-retransmissiontime <positive_integer>]

Arguments

name

The name of the ipsec profile

ikeVersion

IKE Protocol Version

Possible values: V1, V2

encAlgo

Type of encryption algorithm

hashAlgo

Type of hashing algorithm

lifetime

Lifetime of SA in seconds

Minimum value: 60

Maximum value: 31536000

psk

Pre shared key value

publickey

Public key file path

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveliness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveliness checks.

Maximum value: 64999

replayWindowSize

IPSec Replay window size for the data traffic

Maximum value: 16384

ikeRetryInterval

IKE retry interval for bringing up the connection

Minimum value: 60

Maximum value: 3600

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.

Minimum value: 1

Maximum value: 99

---

show ipsec profile

Display all of the configured ipsec peers

Synopsys

show ipsec profile [<name>]

Arguments

name

The name of the ipsec profile

summary

fullValues

format

level

Outputs

ikeVersion

IKE Protocol Version

encAlgo

Type of encryption algorithm.

hashAlgo

Type of hashing algorithm

lifetime

Lifetime of SA in seconds

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveliness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveliness checks.

replayWindowSize

IPSec Replay window size for the data traffic

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.

psk

Pre shared key value

publickey

Public key file path

privatekey

Private key file path

peerPublicKey

Peer public key file path

ikeRetryInterval

IKE retry interval for bringing up the connection

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

Example

show ipsec profile

---

rm ipsec profile

Remove an ipsec peer

Synopsys

rm ipsec profile <name>

Arguments

name

The name of the ipsec profile.

Example

rm ipsec profile

Load Balancing Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- lb group
- lb metricTable
- lb monbindings
- lb monitor
- lb parameter
- lb persistentSessions
- lb route
- lb route6
- lb sipParameters
- lb vserver
- lb wlm

lb group

Sep 22, 2015

The following operations can be performed on "lb group":

set | unset | bind | unbind | show | rename

set lb group

Configures persistence for the specified load balancing group. The persistence settings are applied to all the members of the group.

Synopsys

set lb group <name>@ [-persistenceType <persistenceType>] [-persistenceBackup (SOURCEIP | NONE)] [-backupPersistenceTimeout <mins>] [-persistMask <netmask>] [-cookieName <string>] [-v6persistmasklen <positive_integer>] [-cookieDomain <string>] [-timeout <mins>] [-rule <expression>]

Arguments

name

Name of the load balancing virtual server group.

persistenceType

Type of persistence for the group. Available settings function as follows:

* SOURCEIP - Create persistence sessions based on the client IP.

* COOKIEINSERT - Create persistence sessions based on a cookie in client requests. The cookie is inserted by a Set-Cookie directive from the server, in its first response to a client.

* RULE - Create persistence sessions based on a user defined rule.

* NONE - Disable persistence for the group.

Possible values: SOURCEIP, COOKIEINSERT, RULE, NONE

persistenceBackup

Type of backup persistence for the group.

Possible values: SOURCEIP, NONE

backupPersistenceTimeout

Time period, in minutes, for which backup persistence is in effect.

Default value: 2

Minimum value: 2

Maximum value: 1440

persistMask

Persistence mask to apply to source IPv4 addresses when creating source IP based persistence sessions.

Default value: 0xFFFFFFFF

cookieName

Use this parameter to specify the cookie name for COOKIE peristence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

v6persistmasklen

Persistence mask to apply to source IPv6 addresses when creating source IP based persistence sessions.

Default value: 128

Minimum value: 1

Maximum value: 128

cookieDomain

Domain attribute for the HTTP cookie.

timeout

Time period for which a persistence session is in effect.

Default value: 2

Maximum value: 1440

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "'<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Default value: "None"

Example

set lb group webgrp -persistenceType COOKIEINSERT To set the persistence type for multiple groups use the following command: set lb group webgrp[1-3] -persisten

---

unset lb group

Use this command to remove lb group settings.Refer to the set lb group command for meanings of the arguments.

Synopsys

unset lb group <name>@ [-persistenceType] [-persistenceBackup] [-backupPersistenceTimeout] [-persistMask] [-cookieName] [-v6persistmasklen] [-cookieDomain] [-timeout] [-rule]

---

bind lb group

Binds one or more virtual servers to a load balancing virtual server group. If the specified group does not exist, the NetScaler appliance first creates the group, and then binds the virtual servers to it. A virtual server group enables you to specify common persistence settings for all of its members through a single set lb group command. Only address-based virtual servers can be added to a group. Content-based virtual servers (content switching and cache redirection virtual servers) cannot be added. A virtual server can be assigned to only one group at any given time. To move a virtual server from one group to another, the virtual server must first be unbound from the group to which it belongs.

Synopsys

bind lb group <name>@ <vServerName>@ ...

Arguments

name

Name for the load balancing virtual server group. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my lbgroup" or 'my lbgroup').

vServerName

Name of the virtual server to bind to the group. Multiple names can be specified.

Example

bind lb group webgrp http_vip To bind multiple vservers to a group use the following command: bind lb group webgrp v[1-4] To bind vserver v1 to group webgrp1, v2 to w

---

unbind lb group

Unbinds one or more virtual servers from a group. When the last virtual server is unbound, the group is removed.

Synopsys

unbind lb group <name> <vServerName>@ ...

Arguments

name

Name of the load balancing virtual server group.

vServerName

Name of the virtual server to unbind. Multiple names can be specified.

Example

unbind lb group webgroup http_vip To unbind multiple vservers use the following command: unbind lb group webgroup v[1-3]

---

show lb group

Displays the virtual servers bound to the specified group.

Synopsys

show lb group [<name>]

Arguments

name

Name of the load balancing virtual server group.

summary

fullValues

format

level

Outputs

vServerName

Virtual server name.

persistenceType

The type of the persistence set for the group.

persistenceBackup

The type of the backup persistence set for the group.

backupPersistenceTimeout

Time period, in minutes, for which backup persistence is in effect.

persistMask

The netmask applied for ipv4 traffic when the persistency type is SOURCEIP.

v6persistmasklen

The netmask applied for ipv6 traffic when the persistency type is SOURCEIP.

cookieName

Use this parameter to specify the cookie name for COOKIE peristence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

cookieDomain

Domain attribute for the HTTP cookie.

timeout

Time period for which a persistence session is in effect.

rule

Rule type.

td

Traffic Domain ID

stateflag

devno

count

Example

show lb group webgrp

---

rename lb group

Renames a load balancing virtual server group.

Synopsys

rename lb group <name>@ <newName>@

Arguments

name

Existing name of the load balancing virtual server group.

newName

New name for the load balancing virtual server group.

Example

```
rename lb group gv1 gv-new1
```

lb metricTable

Sep 22, 2015

The following operations can be performed on "lb metricTable":

add | rm | set | bind | unbind | show

add lb metricTable

Creates a metric table for load monitoring.

Synopsys

add lb metricTable <metricTable>

Arguments

metricTable

Name for the metric table. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my metrictable" or 'my metrictable').

Example

```
add metrictable newtable
```

---

rm lb metricTable

Removes a metric table.

Synopsys

rm lb metricTable <metricTable>

Arguments

metricTable

Name of the metric table.

Example

```
rm metric table netscaler
```

---

set lb metricTable

Modifies the SNMP OID of a metric in a metric table.

Synopsys

set lb metricTable <metricTable> <metric> <snmpOID>

Arguments

metricTable

Name of the metric table.

Example

```
set metrictable table met1 aliasname oidstr
```

---

bind lb metricTable

Binds a metric to a metric table. You must also specify the SNMP OID of the metric.

Synopsys

bind lb metricTable <metricTable> <metric> <snmpOID>

Arguments

metricTable

Name of the metric table.

metric

Name of the metric.

Example

bind metrictable tablename aliasname 1.2.3.4

---

unbind lb metricTable

Unbinds a metric from a metric table.

Synopsys

unbind lb metricTable <metricTable> <metric>

Arguments

metricTable

Name of the metric table.

metric

Name of the metric to unbind.

Example

unbind metrictable tablename aliasname

---

show lb metricTable

Displays the parameters of the specified metric table. If no metric table name is specified, a list of all configured metric tables is displayed.

Synopsys

show lb metricTable [<metricTable>]

Arguments

metricTable

Name of the metric table.

summary

fullValues

format

level

Outputs

metric

Metric name of the oid.

snmpOID

OID corresponding to the metric

flags

flags controlling displayNOTE: This attribute is deprecated.This is deprecated attribute.

stateflag

flags controlling display

metricType

Indication if it is a configured or internal

type

Adds a temporary or permanent table.

devno

count

Example

An example of the show metrictable command output is as follows: Name : ALTEON Type : INTERNAL Name : CISCO-

---

lb monbindings

Sep 22, 2015

The following operations can be performed on "lb monbindings":

show lb monbindings

Display the services to which this monitor is bound

Synopsys

show lb monbindings <monitorName>

Arguments

monitorName

The name of the monitor.

summary

fullValues

Outputs

type

The type of monitor.

state

The state of the monitor.

boundServiceGroupSvrState

The state of the servicegroup.

monsvcState

The configured state (enable/disable) of Monitor on this service.

monState

The configured state (enable/disable) of Monitor on this service.

IPAddress

The IPAddress of the service.

port

The port of the service.

serviceName

The name of the service.

serviceGroupName

The name of the service group.

serviceType

The type of service

svrState

The state of the service

stateflag

devno

count

lb monitor

Sep 22, 2015

The following operations can be performed on "lb monitor":

add | rm | set | unset | enable | disable | bind | unbind | show

add lb monitor

Creates a monitor that you can bind to load balancing services. The monitor periodically sends probes to those services to test their availability.

Synopsys

add lb monitor <monitorName> <type> [-action <action>] [-respCode <int[-int]> ...] [-httpRequest <string>] [-rtspRequest <string>] [-customHeaders <string>] [-maxForwards <positive_integer>] [-sipMethod <sipMethod>] [-sipURI <string>] [-sipregURI <string>] [-send <string>] [-recv <string>] [-query <string>] [-queryType <queryType>] [-scriptName <string>] [-scriptArgs <string>] [-dispatcherIP <ip_addr>] [-dispatcherPort <port>] [-userName <string>] {-password } {-secondaryPassword } [-logonpointName <string>] [-lasVersion <string>] {-radKey } [-radNASid <string>] [-radNASip <ip_addr>] [-radAccountType <positive_integer>] [-radFramedIP <ip_addr>] [-radAPN <string>] [-radMSISDN <string>] [-radAccountSession <string>] [-LRTM (ENABLED | DISABLED)] [-deviation <positive_integer> [<units>]] [-interval <integer> [<units>]] [-resptimeout <integer> [<units>]] [-resptimeoutThresh <positive_integer>] [-retries <integer>] [-failureRetries <integer>] [-alertRetries <integer>] [-successRetries <integer>] [-downTime <integer> [<units>]] [-destIP <ip_addr|ipv6_addr>] [-destPort <port>] [-state (ENABLED | DISABLED)] [-reverse (YES | NO)] [-transparent (YES | NO)] [-ipTunnel (YES | NO)] [-tos (YES | NO)] [-tosId <positive_integer>] [-secure (YES | NO)] [-validateCred (YES | NO)] [-domain <string>] [-IPAddress <ip_addr|ipv6_addr|*> ...] [-group <string>] [-fileName <string>] [-baseDN <string>] [-bindDN <string>] [-filter <string>] [-attribute <string>] [-database <string>] [-sqlQuery <text>] [-evalRule <expression>] [-mssqlProtocolVersion <mssqlProtocolVersion>] [-snmpOID <string>] [-snmpCommunity <string>] [-snmpThreshold <string>] [-snmpVersion (V1 | V2)] [-metricTable <string>] [-application <string>] [-sitePath <string>] [-storename <string>] [-storefrontacctservice (YES | NO)] [-netProfile <string>] [-originHost <string>] [-originRealm <string>] [-hostIPAddress <ip_addr|ipv6_addr|*>] [-vendorId <positive_integer>] [-productName <string>] [-firmwareRevision <positive_integer>] [-authApplicationId <positive_integer> ...] [-acctApplicationId <positive_integer> ...] [-inbandSecurityId (NO_INBAND_SECURITY | TLS)] [-supportedVendorIds <positive_integer> ...] [-vendorSpecificVendorId <positive_integer> [-vendorSpecificAuthApplicationIds <positive_integer> ...] [-vendorSpecificAcctApplicationIds <positive_integer> ...]] [-kcdAccount <string>] [-storedb (ENABLED | DISABLED)]

Arguments

monitorName

Name for the monitor. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my monitor" or 'my monitor').

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT

action

Action to perform when the response to an inline monitor (a monitor of type HTTP-INLINE) indicates that the service is down. A service monitored by an inline monitor is considered DOWN if the response code is not one of the codes that have been specified for the Response Code parameter.

Available settings function as follows:

* NONE - Do not take any action. However, the show service command and the show lb monitor command indicate the total number of responses that were checked and the number of consecutive error responses received after the last successful probe.

* LOG - Log the event in NSLOG or SYSLOG.

* DOWN - Mark the service as being down, and then do not direct any traffic to the service until the configured down time has expired. Persistent connections to the service are terminated as soon as the service is marked as DOWN. Also, log the event in NSLOG or SYSLOG.

Possible values: NONE, LOG, DOWN

Default value: SM_DOWN

respCode

Response codes for which to mark the service as UP. For any other response code, the action performed depends on the monitor type. HTTP monitors and RADIUS monitors mark the service as DOWN, while HTTP-INLINE monitors perform the action indicated by the Action parameter.

httpRequest

HTTP request to send to the server (for example, "HEAD /file.html").

rtspRequest

RTSP request to send to the server (for example, "OPTIONS *").

customHeaders

Custom header string to include in the monitoring probes.

maxForwards

Maximum number of hops that the SIP request used for monitoring can traverse to reach the server. Applicable only to monitors of type SIP-UDP.

Default value: 1

Maximum value: 255

sipMethod

SIP method to use for the query. Applicable only to monitors of type SIP-UDP.

Possible values: OPTIONS, INVITE, REGISTER

sipURI

SIP URI string to send to the service (for example, sip:sip.test). Applicable only to monitors of type SIP-UDP.

sipregURI

SIP user to be registered. Applicable only if the monitor is of type SIP-UDP and the SIP Method parameter is set to REGISTER.

send

String to send to the service. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

recv

String expected from the server for the service to be marked as UP. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

query

Domain name to resolve as part of monitoring the DNS service (for example, example.com).

queryType

Type of DNS record for which to send monitoring queries. Set to Address for querying A records, AAAA for querying AAAA records, and Zone for querying the SOA record.

Possible values: Address, Zone, AAAA

scriptName

Path and name of the script to execute. The script must be available on the NetScaler appliance, in the /nsconfig/monitors/ directory.

scriptArgs

String of arguments for the script. The string is copied verbatim into the request.

dispatcherIP

IP address of the dispatcher to which to send the probe.

dispatcherPort

Port number on which the dispatcher listens for the monitoring probe.

userName

User name with which to probe the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, or CITRIX-WI-EXTENDED server.

password

Password that is required for logging on to the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, or CITRIX-WI-EXTENDED server. Used in conjunction with the user name specified for the User Name parameter.

secondaryPassword

Secondary password that users might have to provide to log on to the Access Gateway server. Applicable to CITRIX-AG monitors.

logonpointName

Name of the logon point that is configured for the Citrix Access Gateway Advanced Access Control software. Required if you want to monitor the associated login page or Logon Agent. Applicable to CITRIX-AAC-LAS and CITRIX-AAC-LOGINPAGE monitors.

lasVersion

Version number of the Citrix Advanced Access Control Logon Agent. Required by the CITRIX-AAC-LAS monitor.

radKey

Authentication key (shared secret text string) for RADIUS clients and servers to exchange. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radNASid

NAS-Identifier to send in the Access-Request packet. Applicable to monitors of type RADIUS.

radNASip

Network Access Server (NAS) IP address to use as the source IP address when monitoring a RADIUS server. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radAccountType

Account Type to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

Default value: 1

Maximum value: 15

radFramedIP

Source ip with which the packet will go out . Applicable to monitors of type RADIUS_ACCOUNTING.

radAPN

Called Station Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radMSISDN

Calling Stations Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radAccountSession

Account Session ID to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

LRTM

Calculate the least response times for bound services. If this parameter is not enabled, the appliance does not learn the response times of the bound services. Also used for LRTM load balancing.

Possible values: ENABLED, DISABLED

deviation

Time value added to the learned average response time in dynamic response time monitoring (DRTM). When a deviation is specified, the appliance learns the average response time of bound services and adds the deviation to the average. The final value is then continually adjusted to accommodate response time variations over time. Specified in milliseconds, seconds, or minutes.

Maximum value: 20939000

interval

Time interval between two successive probes. Must be greater than the value of Response Time-out.

Default value: 5

Minimum value: 1

Maximum value: 20940000

resptimeout

Amount of time for which the appliance must wait before it marks a probe as FAILED. Must be less than the value specified for the Interval parameter.

Note: For UDP-ECV monitors for which a receive string is not configured, response timeout does not apply. For UDP-ECV monitors with no receive string, probe failure is indicated by an ICMP port unreachable error received from the service.

Default value: 2

Minimum value: 1

Maximum value: 20939000

resptimeoutThresh

Response time threshold, specified as a percentage of the Response Time-out parameter. If the response to a monitor probe has not arrived when the threshold is reached, the appliance generates an SNMP trap called monRespTimeoutAboveThresh. After the response time returns to a value below the threshold, the appliance generates a monRespTimeoutBelowThresh SNMP trap. For the traps to be generated, the "MONITOR-RTO-THRESHOLD" alarm must also be enabled.

Maximum value: 100

retries

Maximum number of probes to send to establish the state of a service for which a monitoring probe failed.

Default value: 3

Minimum value: 1

Maximum value: 127

failureRetries

Number of retries that must fail, out of the number specified for the Retries parameter, for a service to be marked as DOWN. For example, if the Retries parameter is set to 10 and the Failure Retries parameter is set to 6, out of the ten probes sent, at least six probes must fail if the service is to be marked as DOWN. The default value of 0 means that all the retries must fail if the service is to be marked as DOWN.

Maximum value: 32

alertRetries

Number of consecutive probe failures after which the appliance generates an SNMP trap called monProbeFailed.

Maximum value: 32

successRetries

Number of consecutive successful probes required to transition a service's state from DOWN to UP.

Default value: 1

Minimum value: 1

Maximum value: 32

downTime

Time duration for which to wait before probing a service that has been marked as DOWN. Expressed in milliseconds, seconds, or minutes.

Default value: 30

Minimum value: 1

Maximum value: 20939000

destIP

IP address of the service to which to send probes. If the parameter is set to 0, the IP address of the server to which the monitor is bound is considered the destination IP address.

destPort

TCP or UDP port to which to send the probe. If the parameter is set to 0, the port number of the service to which the monitor is bound is considered the destination port. For a monitor of type USER, however, the destination port is the port number that is included in the HTTP request sent to the dispatcher. Does not apply to monitors of type PING.

state

State of the monitor. The DISABLED setting disables not only the monitor being configured, but all monitors of the same type, until the parameter is set to ENABLED. If the monitor is bound to a service, the state of the monitor is not taken into account when the state of the service is determined.

Possible values: ENABLED, DISABLED

Default value: ENABLED

reverse

Mark a service as DOWN, instead of UP, when probe criteria are satisfied, and as UP instead of DOWN when probe criteria are not satisfied.

Possible values: YES, NO

Default value: NO

transparent

The monitor is bound to a transparent device such as a firewall or router. The state of a transparent device depends on the responsiveness of the services behind it. If a transparent device is being monitored, a destination IP address must be specified. The probe is sent to the specified IP address by using the MAC address of the transparent device.

Possible values: YES, NO

Default value: NO

ipTunnel

Send the monitoring probe to the service through an IP tunnel. A destination IP address must be specified.

Possible values: YES, NO

Default value: NO

tos

Probe the service by encoding the destination IP address in the IP TOS (6) bits.

Possible values: YES, NO

tosId

The TOS ID of the specified destination IP. Applicable only when the TOS parameter is set.

Minimum value: 1

Maximum value: 63

secure

Use a secure SSL connection when monitoring a service. Applicable only to TCP based monitors. The secure option cannot be used with a CITRIX-AG monitor, because a CITRIX-AG monitor uses a secure connection by default.

Possible values: YES, NO

Default value: NO

validateCred

Validate the credentials of the Xen Desktop DDC server user. Applicable to monitors of type CITRIX-XD-DDC.

Possible values: YES, NO

Default value: NO

domain

Domain in which the XenDesktop Desktop Delivery Controller (DDC) servers or Web Interface servers are present. Required by CITRIX-XD-DDC and CITRIX-WI-EXTENDED monitors for logging on to the DDC servers and Web Interface servers, respectively.

IPAddress

Set of IP addresses expected in the monitoring response from the DNS server, if the record type is A or AAAA. Applicable to DNS monitors.

group

Name of a newsgroup available on the NNTP service that is to be monitored. The appliance periodically generates an NNTP query for the name of the newsgroup and evaluates the response. If the newsgroup is found on the server, the service is marked as UP. If the newsgroup does not exist or if the search fails, the service is marked as DOWN. Applicable to NNTP monitors.

fileName

Name of a file on the FTP server. The appliance monitors the FTP service by periodically checking the existence of the file on the server. Applicable to FTP-EXTENDED monitors.

baseDN

The base distinguished name of the LDAP service, from where the LDAP server can begin the search for the attributes in the monitoring query. Required for LDAP service monitoring.

bindDN

The distinguished name with which an LDAP monitor can perform the Bind operation on the LDAP server. Optional. Applicable to LDAP monitors.

filter

Filter criteria for the LDAP query. Optional.

attribute

Attribute to evaluate when the LDAP server responds to the query. Success or failure of the monitoring probe depends on whether the attribute exists in the response. Optional.

database

Name of the database to probe. The name is used to connect to the database during authentication.

sqlQuery

SQL query for a MYSQL-ECV or MSSQL-ECV monitor. Sent to the database server after the server authenticates the connection.

evalRule

Default syntax expression that evaluates the database server's response to a MYSQL-ECV or MSSQL-ECV monitoring query. Must produce a Boolean result. The result determines the state of the server. If the expression returns TRUE, the probe succeeds.

For example, if you want the appliance to evaluate the error message to determine the state of the server, use the rule MYSQL.RES.ROW(10) .TEXT_ELEM(2).EQ("MySQL").

mssqlProtocolVersion

Version of MSSQL server that is to be monitored.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: TDS_PROT_70

snmpOID

SNMP OID for SNMP monitors.

snmpCommunity

Community name for SNMP monitors.

snmpThreshold

Threshold for SNMP monitors.

snmpVersion

SNMP version to be used for SNMP monitors.

Possible values: V1, V2

metricTable

Metric table to which to bind metrics.

application

Name of the application used to determine the state of the service. Applicable to monitors of type CITRIX-XML-SERVICE.

sitePath

URL of the logon page. For monitors of type CITRIX-WEB-INTERFACE, to monitor a dynamic page under the site path, terminate the site path with a slash (/). Applicable to CITRIX-WEB-

INTERFACE and CITRIX-WI-EXTENDED monitors.

storename

Store Name. For monitors of type STOREFRONT, STORENAME is an optional argument defining storefront service store name. Applicable to STOREFRONT monitors.

storefrontacctservice

Enable/Disable probing for Account Service. Applicable only to Store Front monitors. For multi-tenancy configuration users my skip account service

Possible values: YES, NO

Default value: YES

hostName

Hostname in the FQDN format (Example: porche.cars.org). Applicable to STOREFRONT monitors.

netProfile

Name of the network profile.

originHost

Origin-Host value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

originRealm

Origin-Realm value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

hostIPAddress

Host-IP-Address value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. If Host-IP-Address is not specified, the appliance inserts the mapped IP (MIP) address or subnet IP (SNIP) address from which the CER request (the monitoring probe) is sent.

vendorId

Vendor-Id value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

productName

Product-Name value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

firmwareRevision

Firmware-Revision value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

authApplicationId

List of Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring CER message.

Maximum value: 4294967295

acctApplicationId

List of Acct-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message.

Maximum value: 4294967295

inbandSecurityId

Inband-Security-Id for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Possible values: NO_INBAND_SECURITY, TLS

supportedVendorIds

List of Supported-Vendor-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum eight of these AVPs are supported in a monitoring message.

Minimum value: 1

Maximum value: 4294967295

vendorSpecificVendorId

Vendor-Id to use in the Vendor-Specific-Application-Id grouped attribute-value pair (AVP) in the monitoring CER message. To specify Auth-Application-Id or Acct-Application-Id in Vendor-Specific-Application-Id, use vendorSpecificAuthApplicationIds or vendorSpecificAcctApplicationIds, respectively. Only one Vendor-Id is supported for all the Vendor-Specific-Application-Id AVPs in a CER monitoring message.

Minimum value: 1

kcdAccount

KCD Account used by MSSQL monitor

storedb

Used in case of DB specific LB.If enalbed then we store the database list populated from monitors responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

add monitor http_mon http

---

rm lb monitor

Removes a monitor or a response code for an HTTP monitor. If you do not specify any response codes, the monitor is removed. If you provide any or all of the HTTP response codes that are configured for the monitor, only those specified response codes are removed; the monitor is not removed. Built-in monitors cannot be removed.

Synopsys

rm lb monitor <monitorName> <type> [-respCode <int[-int]> ...]

Arguments

monitorName

Name of the monitor.

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT

respCode

Response codes to delete from the response code list configured for the HTTP monitor.

Example

rm monitor http_mon http

---

set lb monitor

Modifies the specified parameters of a monitor.

Synopsys

set lb monitor <monitorName> <type> [-action <action>] [-respCode <int[-int]> ...] [-httpRequest <string>] [-rtspRequest <string>] [-customHeaders <string>] [-maxForwards <positive_integer>] [-sipMethod <sipMethod>] [-sipregURI <string>] [-sipURI <string>] [-send <string>] [-recv <string>] [-query <string>] [-queryType <queryType>] [-userName <string>] {-password } {-secondaryPassword } [-logonpointName <string>] [-lasVersion <string>] {-radKey } [-radNASid <string>] [-radNASip <ip_addr>] [-radAccountType <positive_integer>] [-radFramedIP <ip_addr>] [-radAPN <string>] [-radMSISDN <string>] [-radAccountSession <string>] [-LRTM (ENABLED | DISABLED)] [-deviation <positive_integer> [<units>]] [-scriptName <string>] [-scriptArgs <string>] [-validateCred (YES | NO)] [-domain <string>] [-dispatcherIP <ip_addr>] [-dispatcherPort <port>] [-interval <integer> [<units>]] [-resptimeout <integer> [<units>]] [-resptimeoutThresh <positive_integer>] [-retries <integer>] [-failureRetries <integer>] [-alertRetries <integer>] [-successRetries <integer>] [-downTime <integer> [<units>]] [-destIP <ip_addr|ipv6_addr>] [-destPort <port>] [-state (ENABLED | DISABLED)] [-reverse (YES | NO)] [-transparent (YES | NO)] [-ipTunnel (YES | NO)] [-tos (YES | NO)] [-tosId <positive_integer>] [-secure (YES | NO)] [-IPAddress <ip_addr|ipv6_addr|*> ...] [-group <string>] [-fileName <string>] [-baseDN <string>] [-bindDN <string>] [-filter <string>] [-attribute <string>] [-database <string>] [-sqlQuery <text>] [-evalRule <expression>] [-snmpOID <string>] [-snmpCommunity <string>] [-snmpThreshold <string>] [-snmpVersion (V1 | V2)] [-metricTable <string>] [-metric <string> [-metricThreshold <positive_integer>] [-metricWeight <positive_integer>]] [-application <string>] [-sitePath <string>] [-storename <string>] [-storefrontacctservice (YES | NO)] [-netProfile <string>] [-mssqlProtocolVersion <mssqlProtocolVersion>] [-originHost <string>] [-originRealm <string>] [-hostIPAddress <ip_addr|ipv6_addr|*>] [-vendorId <positive_integer>] [-productName <string>] [-firmwareRevision <positive_integer>] [-authApplicationId <positive_integer> ...] [-acctApplicationId <positive_integer> ...] [-inbandSecurityId (NO_INBAND_SECURITY | TLS)] [-supportedVendorIds <positive_integer> ...] [-vendorSpecificVendorId <positive_integer> [-vendorSpecificAuthApplicationIds <positive_integer> ...] [-vendorSpecificAcctApplicationIds <positive_integer> ...]] [-kcdAccount <string>]

Arguments

monitorName

Name of the monitor.

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT

action

Action to perform when the response to an inline monitor (a monitor of type HTTP-INLINE) indicates that the service is down. A service monitored by an inline monitor is considered DOWN if the response code is not one of the codes that have been specified for the Response Code parameter.

Available settings function as follows:

* NONE - Do not take any action. However, the show service command and the show lb monitor command indicate the total number of responses that were checked and the number of

consecutive error responses received after the last successful probe.

* LOG - Log the event in NSLOG or SYSLOG.

* DOWN - Mark the service as being down, and then do not direct any traffic to the service until the configured down time has expired. Persistent connections to the service are terminated as soon as the service is marked as DOWN. Also, log the event in NSLOG or SYSLOG.

Possible values: NONE, LOG, DOWN

Default value: SM_DOWN

respCode

Response codes for which to mark the service as UP. For any other response code, the action performed depends on the monitor type. HTTP monitors and RADIUS monitors mark the service as DOWN, while HTTP-INLINE monitors perform the action indicated by the Action parameter.

httpRequest

HTTP request to send to the server (for example, "HEAD /file.html").

rtspRequest

RTSP request to send to the server (for example, "OPTIONS *").

customHeaders

Custom header string to include in the monitoring probes.

maxForwards

Maximum number of hops that the SIP request used for monitoring can traverse to reach the server. Applicable only to monitors of type SIP-UDP.

Default value: 1

Maximum value: 255

sipMethod

SIP method to use for the query. Applicable only to monitors of type SIP-UDP.

Possible values: OPTIONS, INVITE, REGISTER

sipURI

SIP URI string to send to the service (for example, sip:sip.test). Applicable only to monitors of type SIP-UDP.

send

String to send to the service. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

recv

String expected from the server for the service to be marked as UP. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

query

Domain name to resolve as part of monitoring the DNS service (for example, example.com).

queryType

Type of DNS record for which to send monitoring queries. Set to Address for querying A records, AAAA for querying AAAA records, and Zone for querying the SOA record.

Possible values: Address, Zone, AAAA

userName

User name with which to probe the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, or CITRIX-WI-EXTENDED server.

password

Password that is required for logging on to the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, or CITRIX-WI-EXTENDED server. Used in conjunction with the user name specified for the User Name parameter.

secondaryPassword

Secondary password that users might have to provide to log on to the Access Gateway server. Applicable to CITRIX-AG monitors.

logonpointName

Name of the logon point that is configured for the Citrix Access Gateway Advanced Access Control software. Required if you want to monitor the associated login page or Logon Agent. Applicable to CITRIX-AAC-LAS and CITRIX-AAC-LOGINPAGE monitors.

lasVersion

Version number of the Citrix Advanced Access Control Logon Agent. Required by the CITRIX-AAC-LAS monitor.

radKey

Authentication key (shared secret text string) for RADIUS clients and servers to exchange. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radNASid

NAS-Identifier to send in the Access-Request packet. Applicable to monitors of type RADIUS.

radNASip

Network Access Server (NAS) IP address to use as the source IP address when monitoring a RADIUS server. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radAccountType

Account Type to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

Default value: 1

Maximum value: 15

radFramedIP

Source ip with which the packet will go out . Applicable to monitors of type RADIUS_ACCOUNTING.

radAPN

Called Station Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radMSISDN

Calling Stations Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radAccountSession

Account Session ID to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

LRTM

Calculate the least response times for bound services. If this parameter is not enabled, the appliance does not learn the response times of the bound services. Also used for LRTM load balancing.

Possible values: ENABLED, DISABLED

deviation

Time value added to the learned average response time in dynamic response time monitoring (DRTM). When a deviation is specified, the appliance learns the average response time of bound services and adds the deviation to the average. The final value is then continually adjusted to accommodate response time variations over time. Specified in milliseconds, seconds, or minutes.

Maximum value: 20939000

scriptName

Path and name of the script to execute. The script must be available on the NetScaler appliance, in the /nsconfig/monitors/ directory.

scriptArgs

String of arguments for the script. The string is copied verbatim into the request.

validateCred

Validate the credentials of the Xen Desktop DDC server user. Applicable to monitors of type CITRIX-XD-DDC.

Possible values: YES, NO

Default value: NO

domain

Domain in which the XenDesktop Desktop Delivery Controller (DDC) servers or Web Interface servers are present. Required by CITRIX-XD-DDC and CITRIX-WI-EXTENDED monitors for logging on to the DDC servers and Web Interface servers, respectively.

dispatcherIP

IP address of the dispatcher to which to send the probe.

dispatcherPort

Port number on which the dispatcher listens for the monitoring probe.

interval

Time interval between two successive probes. Must be greater than the value of Response Time-out.

Default value: 5

Minimum value: 1

Maximum value: 20940000

resptimeout

Amount of time for which the appliance must wait before it marks a probe as FAILED. Must be less than the value specified for the Interval parameter.

Note: For UDP-ECV monitors for which a receive string is not configured, response timeout does not apply. For UDP-ECV monitors with no receive string, probe failure is indicated by an ICMP

port unreachable error received from the service.

Default value: 2

Minimum value: 1

Maximum value: 20939000

resptimeoutThresh

Response time threshold, specified as a percentage of the Response Time-out parameter. If the response to a monitor probe has not arrived when the threshold is reached, the appliance generates an SNMP trap called monRespTimeoutAboveThresh. After the response time returns to a value below the threshold, the appliance generates a monRespTimeoutBelowThresh SNMP trap. For the traps to be generated, the "MONITOR-RTO-THRESHOLD" alarm must also be enabled.

Maximum value: 100

retries

Maximum number of probes to send to establish the state of a service for which a monitoring probe failed.

Default value: 3

Minimum value: 1

Maximum value: 127

failureRetries

Number of retries that must fail, out of the number specified for the Retries parameter, for a service to be marked as DOWN. For example, if the Retries parameter is set to 10 and the Failure Retries parameter is set to 6, out of the ten probes sent, at least six probes must fail if the service is to be marked as DOWN. The default value of 0 means that all the retries must fail if the service is to be marked as DOWN.

Maximum value: 32

alertRetries

Number of consecutive probe failures after which the appliance generates an SNMP trap called monProbeFailed.

Maximum value: 32

successRetries

Number of consecutive successful probes required to transition a service's state from DOWN to UP.

Default value: 1

Minimum value: 1

Maximum value: 32

downTime

Time duration for which to wait before probing a service that has been marked as DOWN. Expressed in milliseconds, seconds, or minutes.

Default value: 30

Minimum value: 1

Maximum value: 20939000

destIP

IP address of the service to which to send probes. If the parameter is set to 0, the IP address of the server to which the monitor is bound is considered the destination IP address.

destPort

TCP or UDP port to which to send the probe. If the parameter is set to 0, the port number of the service to which the monitor is bound is considered the destination port. For a monitor of type USER, however, the destination port is the port number that is included in the HTTP request sent to the dispatcher. Does not apply to monitors of type PING.

state

State of the monitor. The DISABLED setting disables not only the monitor being configured, but all monitors of the same type, until the parameter is set to ENABLED. If the monitor is bound to a service, the state of the monitor is not taken into account when the state of the service is determined.

Possible values: ENABLED, DISABLED

Default value: ENABLED

reverse

Mark a service as DOWN, instead of UP, when probe criteria are satisfied, and as UP instead of DOWN when probe criteria are not satisfied.

Possible values: YES, NO

Default value: NO

transparent

The monitor is bound to a transparent device such as a firewall or router. The state of a transparent device depends on the responsiveness of the services behind it. If a transparent device is

being monitored, a destination IP address must be specified. The probe is sent to the specified IP address by using the MAC address of the transparent device.

Possible values: YES, NO

Default value: NO

ipTunnel

Send the monitoring probe to the service through an IP tunnel. A destination IP address must be specified.

Possible values: YES, NO

Default value: NO

tos

Probe the service by encoding the destination IP address in the IP TOS (6) bits.

Possible values: YES, NO

tosId

The TOS ID of the specified destination IP. Applicable only when the TOS parameter is set.

Minimum value: 1

Maximum value: 63

secure

Use a secure SSL connection when monitoring a service. Applicable only to TCP based monitors. The secure option cannot be used with a CITRIX-AG monitor, because a CITRIX-AG monitor uses a secure connection by default.

Possible values: YES, NO

Default value: NO

IPAddress

Set of IP addresses expected in the monitoring response from the DNS server, if the record type is A or AAAA. Applicable to DNS monitors.

group

Name of a newsgroup available on the NNTP service that is to be monitored. The appliance periodically generates an NNTP query for the name of the newsgroup and evaluates the response. If the newsgroup is found on the server, the service is marked as UP. If the newsgroup does not exist or if the search fails, the service is marked as DOWN. Applicable to NNTP monitors.

fileName

Name of a file on the FTP server. The appliance monitors the FTP service by periodically checking the existence of the file on the server. Applicable to FTP-EXTENDED monitors.

baseDN

The base distinguished name of the LDAP service, from where the LDAP server can begin the search for the attributes in the monitoring query. Required for LDAP service monitoring.

bindDN

The distinguished name with which an LDAP monitor can perform the Bind operation on the LDAP server. Optional. Applicable to LDAP monitors.

filter

Filter criteria for the LDAP query. Optional.

attribute

Attribute to evaluate when the LDAP server responds to the query. Success or failure of the monitoring probe depends on whether the attribute exists in the response. Optional.

database

Name of the database to probe. The name is used to connect to the database during authentication.

sqlQuery

SQL query for a MYSQL-ECV or MSSQL-ECV monitor. Sent to the database server after the server authenticates the connection.

evalRule

Default syntax expression that evaluates the database server's response to a MYSQL-ECV or MSSQL-ECV monitoring query. Must produce a Boolean result. The result determines the state of the server. If the expression returns TRUE, the probe succeeds.

For example, if you want the appliance to evaluate the error message to determine the state of the server, use the rule MYSQL.RES.ROW(10) .TEXT_ELEM(2).EQ("MySQL").

snmpOID

SNMP OID for SNMP monitors.

snmpCommunity

Community name for SNMP monitors.

snmpThreshold

Threshold for SNMP monitors.

snmpVersion

SNMP version to be used for SNMP monitors.

Possible values: V1, V2

metricTable

Metric table to which to bind metrics.

metric

Metric name in the metric table, whose setting is changed. A value zero disables the metric and it will not be used for load calculation

application

Name of the application used to determine the state of the service. Applicable to monitors of type CITRIX-XML-SERVICE.

sitePath

URL of the logon page. For monitors of type CITRIX-WEB-INTERFACE, to monitor a dynamic page under the site path, terminate the site path with a slash (/). Applicable to CITRIX-WEB-INTERFACE and CITRIX-WI-EXTENDED monitors.

storename

Store Name. For monitors of type STOREFRONT, STORENAME is an optional argument defining storefront service store name. Applicable to STOREFRONT monitors.

storefrontacctservice

Enable/Disable probing for Account Service. Applicable only to Store Front monitors. For multi-tenancy configuration users my skip account service

Possible values: YES, NO

Default value: YES

hostName

Hostname in the FQDN format (Example: porche.cars.org). Applicable to STOREFRONT monitors.

netProfile

Name of the network profile.

mssqlProtocolVersion

Version of MSSQL server that is to be monitored.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: TDS_PROT_70

originHost

Origin-Host value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

originRealm

Origin-Realm value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

hostIPAddress

Host-IP-Address value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. If Host-IP-Address is not specified, the appliance inserts the mapped IP (MIP) address or subnet IP (SNIP) address from which the CER request (the monitoring probe) is sent.

vendorId

Vendor-Id value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

productName

Product-Name value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

firmwareRevision

Firmware-Revision value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

authApplicationId

List of Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring CER message.

Maximum value: 4294967295

acctApplicationId

List of Acct-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are

supported in a monitoring message.

Maximum value: 4294967295

inbandSecurityId

Inband-Security-Id for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Possible values: NO_INBAND_SECURITY, TLS

supportedVendorIds

List of Supported-Vendor-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum eight of these AVPs are supported in a monitoring message.

Minimum value: 1

Maximum value: 4294967295

vendorSpecificVendorId

Vendor-Id to use in the Vendor-Specific-Application-Id grouped attribute-value pair (AVP) in the monitoring CER message. To specify Auth-Application-Id or Acct-Application-Id in Vendor-Specific-Application-Id, use vendorSpecificAuthApplicationIds or vendorSpecificAcctApplicationIds, respectively. Only one Vendor-Id is supported for all the Vendor-Specific-Application-Id AVPs in a CER monitoring message.

Minimum value: 1

kcdAccount

KCD Account used by MSSQL monitor

Example

```
set monitor http_mon http -respcode 100
```

---

unset lb monitor

Removes the specified parameter settings from the specified monitor. Attributes for which a default value is available revert to their default values..Refer to the set lb monitor command for meanings of the arguments.

Synopsys

```
unset lb monitor <monitorName> <type> [-IPAddress <ip_addr|ipv6_addr|*> ...] [-scriptName] [-destPort] [-netProfile] [-action] [-respCode] [-httpRequest] [-rtspRequest] [-customHeaders] [-maxForwards] [-sipMethod] [-sipregURI] [-send] [-recv] [-query] [-queryType] [-userName] [-password] [-secondaryPassword] [-logonpointName] [-lasVersion] [-radKey] [-radNASid] [-radNASip] [-radAccountType] [-radFramedIP] [-radAPN] [-radMSISDN] [-radAccountSession] [-LRTM] [-deviation] [-scriptArgs] [-validateCred] [-domain] [-dispatcherIP] [-dispatcherPort] [-interval] [-resptimeout] [-resptimeoutThresh] [-retries] [-failureRetries] [-alertRetries] [-successRetries] [-downTime] [-destIP] [-state] [-reverse] [-transparent] [-ipTunnel] [-tos] [-tosId] [-secure] [-group] [-fileName] [-baseDN] [-bindDN] [-filter] [-attribute] [-database] [-sqlQuery] [-evalRule] [-snmpOID] [-snmpCommunity] [-snmpThreshold] [-snmpVersion] [-metricTable] [-mssqlProtocolVersion] [-originHost] [-originRealm] [-hostIPAddress] [-vendorId] [-productName] [-firmwareRevision] [-authApplicationId] [-acctApplicationId] [-inbandSecurityId] [-supportedVendorIds] [-vendorSpecificVendorId] [-vendorSpecificAuthApplicationIds] [-vendorSpecificAcctApplicationIds] [-kcdAccount]
```

Example

```
set monitor dns_mon dns -ipaddress 10.102.27.230
```

---

enable lb monitor

Enable the monitor that is bound to a specific service. If no monitor name is specified, all monitors bound to the service are enabled.

Synopsys

```
enable lb monitor (<serviceName>@ | <serviceGroupName>@) [<monitorName>]
```

Arguments

serviceName

The name of the service to which the monitor is bound.

serviceGroupName

The name of the service group to which the monitor is to be bound.

monitorName

Name for the monitor. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my monitor" or 'my monitor').

Example

```
enable monitor http_svc http_mon  To enable monitor for multiple services use the following command:  enable monitor http_svc[1-3] http_mon
```

---

disable lb monitor

Disable the monitor for a service. If the monitor name is not specified, all monitors bound to the service are disabled.

Synopsys

disable lb monitor (<serviceName>@ | <serviceGroupName>@) [<monitorName>]

Arguments

serviceName

The name of the service being monitored.

serviceGroupName

The name of the service group being monitored.

monitorName

Name for the monitor. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my monitor" or 'my monitor').

Example

disable monitor http_svc http_mon To disable a monitor on multiple services use the following command: disable monitor http_svc[1-3] http_mon

---

bind lb monitor

Binds a monitor to a service or service group. Multiple monitors can be bound to a service or service group.

Synopsys

bind lb monitor <monitorName> [-state (ENABLED | DISABLED)] [-weight <positive_integer>] [-state (ENABLED | DISABLED)] [-weight <positive_integer>] [-metric <string> -metricThreshold <positive_integer> [-metricWeight <positive_integer>]]

Arguments

monitorName

Name of the monitor.

serviceName

Name of the service or service group.

serviceGroupName

Name of the service group.

metric

Name of the metric to be polled by the monitor.

Example

bind monitor http_mon http_svc To bind a monitor to multiple services use the following command: bind monitor http_mon http_svc[1-3]

---

unbind lb monitor

Unbinds a monitor from a service or service group.

Synopsys

unbind lb monitor <monitorName> -metric <string>

Arguments

monitorName

Name of the monitor.

serviceName

Name of the service or service group.

serviceGroupName

Name of the service group.

metric

Name of the metric to be polled by the monitor.

Example

unbind monitor http_mon http_svc To unbind a monitor to multiple services use the following command: unbind monitor http_mon http_svc[1-3]

---

show lb monitor

Displays the parameters of all the monitors configured on the appliance, or the parameters of the specified monitor.

Synopsys

show lb monitor [<monitorName>] [<type>] show lb monitor bindings - alias for 'show lb monbindings'

Arguments

monitorName

Name of the monitor.

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT

summary

fullValues

format

level

Outputs

interval

The frequency at which the probe is sent to the service.

units

Giving the unit of the metric

resptimeout

The interval for which the system waits before it marks the probe as FAILED.

resptimeoutThresh

Response time threshold, specified as a percentage of the Response Time-out parameter. If the response to a monitor probe has not arrived when the threshold is reached, the appliance generates an SNMP trap called monRespTimeoutAboveThresh. After the response time returns to a value below the threshold, the appliance generates a monRespTimeoutBelowThresh SNMP trap. For the traps to be generated, the "MONITOR-RTO-THRESHOLD" alarm must also be enabled.

retries

Maximum number of probes to send to establish the state of a service for which a monitoring probe failed.

failureRetries

Number of retries that must fail, out of the number specified for the Retries parameter, for a service to be marked as DOWN. For example, if the Retries parameter is set to 10 and the Failure Retries parameter is set to 6, out of the ten probes sent, at least six probes must fail if the service is to be marked as DOWN. The default value of 0 means that all the retries must fail if the service is to be marked as DOWN.

alertRetries

The number of failures after which the system generates a SNMP trap.

successRetries

Number of consecutive successful probes required to transition a service's state from DOWN to UP.

downTime

The duration in seconds for which the system waits to make the next probe once the service is marked as DOWN.

destIP

The IP address to which the probe is sent.

destPort

The TCP/UDP port to which the probe is sent.

state

reverse

Mark a service as DOWN, instead of UP, when probe criteria are satisfied, and as UP instead of DOWN when probe criteria are not satisfied.

transparent

The state of the monitor for transparent devices.

ipTunnel

The state of the monitor for tunneled devices.

tos

TOS setting.

tosId

TOS ID

secure

The state of the secure monitoring of services.

action

Action to perform when the response to an inline monitor (a monitor of type HTTP-INLINE) indicates that the service is down. A service monitored by an inline monitor is considered DOWN if the response code is not one of the codes that have been specified for the Response Code parameter.

Available settings function as follows:

* NONE - Do not take any action. However, the show service command and the show lb monitor command indicate the total number of responses that were checked and the number of consecutive error responses received after the last successful probe.

* LOG - Log the event in NSLOG or SYSLOG.

* DOWN - Mark the service as being down, and then do not direct any traffic to the service until the configured down time has expired. Persistent connections to the service are terminated as soon as the service is marked as DOWN. Also, log the event in NSLOG or SYSLOG.

respCode

The response codes.

httpRequest

The HTTP request that is sent to the server.

rtspRequest

The RTSP request that is sent to the server.

send

The string that is sent to the service.

recv

The string that is expected from the server to mark the server as UP.

query

Domain name to resolve as part of monitoring the DNS service (for example, example.com).

queryType

Type of DNS record for which to send monitoring queries. Set to Address for querying A records, AAAA for querying AAAA records, and Zone for querying the SOA record.

userName

Username on the RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3/CITRIX-XD-DDC/CITRIX-WI-EXTENDED server. This user name is used in the probe.

password

Password used in RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3/LDAP/CITRIX-XD-DDC/CITRIX-WI-EXTENDED server monitoring.

secondaryPassword

Secondary password that users might have to provide to log on to the Access Gateway server. Applicable to CITRIX-AG monitors.

logonpointName

Logonpoint name used in Citrix AAC login page monitoring.

lasVersion

Version number of the Citrix Advanced Access Control Logon Agent. Required by the CITRIX-AAC-LAS monitor.

validateCred

Validate the credentials of the Xen Desktop DDC server user. Applicable to monitors of type CITRIX-XD-DDC.

domain

Domain in which the XenDesktop Desktop Delivery Controller (DDC) servers or Web Interface servers are present. Required by CITRIX-XD-DDC and CITRIX-WI-EXTENDED monitors for logging on to the DDC servers and Web Interface servers, respectively.

radKey

Authentication key (shared secret text string) for RADIUS clients and servers to exchange. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radNASid

NAS-Identifier to send in the Access-Request packet. Applicable to monitors of type RADIUS.

radNASip

Network Access Server (NAS) IP address to use as the source IP address when monitoring a RADIUS server. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radAccountType

Account Type to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radFramedIP

Source ip with which the packet will go out . Applicable to monitors of type RADIUS_ACCOUNTING.

radAPN

Called Station Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radMSISDN

Calling Stations Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radAccountSession

Account Session ID to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

LRTM

Calculate the least response times for bound services. If this parameter is not enabled, the appliance does not learn the response times of the bound services. Also used for LRTM load balancing.

lrtmConf

State of LRTM configuration on the monitor.

lrtmConfStr

State of LRTM configuration on the monitor as STRING.

deviation

Deviation from the learnt response time for Dynamic Response Time monitoring.

dynamicResponseTimeout

Response timeout of the DRTM enabled monitor , calculated dynamically based on the history and current response time.

dynamicInterval

Interval between monitoring probes for DRTM enabled monitor , calculated dynamically based monitor response time.

scriptName

Path and name of the script to execute. The script must be available on the NetScaler appliance, in the /nsconfig/monitors/ directory.

scriptArgs

String of arguments for the script. The string is copied verbatim into the request.

dispatcherIP

IP address of the dispatcher to which to send the probe.

dispatcherPort

Port number on which the dispatcher listens for the monitoring probe.

sipURI

SIP URI string to send to the service (for example, sip:sip.test). Applicable only to monitors of type SIP-UDP.

sipMethod

Specifies SIP method to be used for the query

maxForwards

Maximum number of hops a sip monitor packet can go.

sipregURI

Specifies SIP user to be registered

customHeaders

The string that is sent to the service. Applicable to HTTP ,HTTP-ECV and RTSP monitor types.

IPAddress

Set of IP addresses expected in the monitoring response from the DNS server, if the record type is A or AAAA. Applicable to DNS monitors.

group

Name of a newsgroup available on the NNTP service that is to be monitored. The appliance periodically generates an NNTP query for the name of the newsgroup and evaluates the response. If the newsgroup is found on the server, the service is marked as UP. If the newsgroup does not exist or if the search fails, the service is marked as DOWN. Applicable to NNTP monitors.

fileName

Name of a file on the FTP server. The appliance monitors the FTP service by periodically checking the existence of the file on the server. Applicable to FTP-EXTENDED monitors.

baseDN

The base distinguished name of the LDAP service, from where the LDAP server can begin the search for the attributes in the monitoring query. Required for LDAP service monitoring.

bindDN

The distinguished name with which an LDAP monitor can perform the Bind operation on the LDAP server. Optional. Applicable to LDAP monitors.

filter

Filter criteria for the LDAP query. Optional.

attribute

Attribute to evaluate when the LDAP server responds to the query. Success or failure of the monitoring probe depends on whether the attribute exists in the response. Optional.

database

Name of the database to probe. The name is used to connect to the database during authentication.

sqlQuery

SQL query for a MYSQL-ECV or MSSQL-ECV monitor. Sent to the database server after the server authenticates the connection.

evalRule

Default syntax expression that evaluates the database server's response to a MYSQL-ECV or MSSQL-ECV monitoring query. Must produce a Boolean result. The result determines the state of the server. If the expression returns TRUE, the probe succeeds.

For example, if you want the appliance to evaluate the error message to determine the state of the server, use the rule MYSQL.RES.ROW(10) .TEXT_ELEM(2).EQ("MySQL").

snmpOID

SNMP OID for SNMP monitors.

snmpCommunity

Community name for SNMP monitors.

snmpThreshold

Threshold for SNMP monitors.

snmpVersion

SNMP version to be used for SNMP monitoring.

metric

Metric name in the metric table, whose setting is changed

metricTable

Metric table, whose setting is changed

multimetrictable

Metric table to which to bind metrics, to be used only for output purposes.

metricThreshold

Threshold to be used for that metric.

metricWeight

The weight for the specified service metric with respect to others.

stateflag

Flags controlling the display.

flags

Used by build-in monitors.

application

Name of the application used to determine the state of the service. Applicable to monitors of type CITRIX-XML-SERVICE.

sitePath

URL of the logon page. For monitors of type CITRIX-WEB-INTERFACE, to monitor a dynamic page under the site path, terminate the site path with a slash (/). Applicable to CITRIX-WEB-INTERFACE and CITRIX-WI-EXTENDED monitors.

storename

Store Name. For monitors of type STOREFRONT, STORENAME is an optional argument defining storefront service store name. Applicable to STOREFRONT monitors.

storefrontacctservice

Enable/Disable probing for Account Service. Applicable only to Store Front monitors. For multi-tenancy configuration users my skip account service

hostName

Hostname in the FQDN format (Example: porche.cars.org). Applicable to STOREFRONT monitors.NOTE: This attribute is deprecated.This is deprecated attribute.

netProfile

Name of the network profile.

mssqlProtocolVersion

Version of MSSQL server that is to be monitored.

originHost

Origin-Host value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

originRealm

Origin-Realm value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

hostIPAddress

Host-IP-Address value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. If Host-IP-Address is not specified, the appliance inserts the mapped IP (MIP) address or subnet IP (SNIP) address from which the CER request (the monitoring probe) is sent.

vendorId

Vendor-Id value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

productName

Product-Name value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

firmwareRevision

Firmware-Revision value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

authApplicationId

List of Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring CER message.

acctApplicationId

List of Acct-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message.

inbandSecurityId

Inband-Security-Id for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

supportedVendorIds

List of Supported-Vendor-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum eight of these AVPs are supported in a monitoring message.

vendorSpecificVendorId

Vendor-Id to use in the Vendor-Specific-Application-Id grouped attribute-value pair (AVP) in the monitoring CER message. To specify Auth-Application-Id or Acct-Application-Id in Vendor-Specific-Application-Id, use vendorSpecificAuthApplicationIds or vendorSpecificAcctApplicationIds, respectively. Only one Vendor-Id is supported for all the Vendor-Specific-Application-Id AVPs in a CER monitoring message.

vendorSpecificAuthApplicationIds

List of Vendor-Specific-Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

vendorSpecificAcctApplicationIds

List of Vendor-Specific-Acct-Application-Id attribute value pairs (AVPs) to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

serviceName

weight

serviceGroupName

kcdAccount

KCD Account used by MSSQL monitor

storedb

Used in case of DB specific LB.If enalbed then we store the database list populated from monitors responses.

devno

count

Example

An example of the show monitor command output is as follows: 8 configured monitors: 1) Name.....: ping

lb parameter

Sep 22, 2015

The following operations can be performed on "lb parameter":

set | unset | show

set lb parameter

Modifies the specified global load balancing parameters.

Synopsys

set lb parameter [-httpOnlyCookieFlag (ENABLED | DISABLED)] [-consolidatedLConn (YES | NO)] [-usePortForHashLb (YES | NO)] [-preferDirectRoute (YES | NO)] [-startupRRFactor <positive_integer>] [-monitorSkipMaxClient (ENABLED | DISABLED)] [-monitorConnectionClose (RESET | FIN)] [-vServerSpecificMac (ENABLED | DISABLED)]

Arguments

httpOnlyCookieFlag

Include the HttpOnly attribute in persistence cookies. The HttpOnly attribute limits the scope of a cookie to HTTP requests and helps mitigate the risk of cross-site scripting attacks.

Possible values: ENABLED, DISABLED

Default value: ENABLED

consolidatedLConn

To find the service with the fewest connections, the virtual server uses the consolidated connection statistics from all the packet engines. The NO setting allows consideration of only the number of connections on the packet engine that received the new connection.

Possible values: YES, NO

Default value: YES

usePortForHashLb

Include the port number of the service when creating a hash for hash based load balancing methods. With the NO setting, only the IP address of the service is considered when creating a hash.

Possible values: YES, NO

Default value: YES

preferDirectRoute

Perform route lookup for traffic received by the NetScaler appliance, and forward the traffic according to configured routes. Do not set this parameter if you want a wildcard virtual server to direct packets received by the appliance to an intermediary device, such as a firewall, even if their destination is directly connected to the appliance. Route lookup is performed after the packets have been processed and returned by the intermediary

device.

Possible values: YES, NO

Default value: YES

startupRRFactor

Number of requests, per service, for which to apply the round robin load balancing method before switching to the configured load balancing method, thus allowing services to ramp up gradually to full load. Until the specified number of requests is distributed, the NetScaler appliance is said to be implementing the slow start mode (or startup round robin). Implemented for a virtual server when one of the following is true:

* The virtual server is newly created.

* One or more services are newly bound to the virtual server.

* One or more services bound to the virtual server are enabled.

* The load balancing method is changed.

This parameter applies to all the load balancing virtual servers configured on the NetScaler appliance, except for those virtual servers for which the virtual server-level slow start parameters (New Service Startup Request Rate and Increment Interval) are configured. If the global slow start parameter and the slow start parameters for a given virtual server are not set, the appliance implements a default slow start for the virtual server, as follows:

* For a newly configured virtual server, the appliance implements slow start for the first 100 requests received by the virtual server.

* For an existing virtual server, if one or more services are newly bound or newly enabled, or if the load balancing method is changed, the appliance dynamically computes the number of requests for which to implement startup round robin. It obtains this number by multiplying the request rate by the number of bound services (it includes services that are marked as DOWN). For example, if the current request rate is 20 requests/s and ten services are bound to the virtual server, the appliance performs startup round robin for 200 requests.

Not applicable to a virtual server for which a hash based load balancing method is configured.

monitorSkipMaxClient

When a monitor initiates a connection to a service, do not check to determine whether the number of connections to the service has reached the limit specified by the service's Max Clients setting. Enables monitoring to continue even if the service has reached its connection limit.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitorConnectionClose

Close monitoring connections by sending the service a connection termination message with the specified bit set.

Possible values: RESET, FIN

Default value: FIN

vServerSpecificMac

Allow a MAC-mode virtual server to accept traffic returned by an intermediary device, such as a firewall, to which the traffic was previously forwarded by another MAC-mode virtual server. The second virtual server can then distribute that traffic across the destination server farm. Also useful when load balancing Branch Repeater appliances.

Note: The second virtual server can also send the traffic to another set of intermediary devices, such as another set of firewalls. If necessary, you can configure multiple MAC-mode virtual servers to pass traffic successively through multiple sets of intermediary devices.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

set lb parameter -httponly (ENABLED|DISABLED)

---

unset lb parameter

Use this command to remove lb parameter settings.Refer to the set lb parameter command for meanings of the arguments.

Synopsys

unset lb parameter [-httpOnlyCookieFlag] [-consolidatedLConn] [-usePortForHashLb] [-preferDirectRoute] [-startupRRFactor] [-monitorSkipMaxClient] [-monitorConnectionClose] [-vServerSpecificMac]

---

show lb parameter

Displays the global load balancing parameters.

Synopsys

show lb parameter

Arguments

format

level

Outputs

httpOnlyCookieFlag

Include the HttpOnly attribute in persistence cookies. The HttpOnly attribute limits the scope of a cookie to HTTP requests and helps mitigate the risk of cross-site scripting attacks.

consolidatedLConn

To find the service with the fewest connections, the virtual server uses the consolidated connection statistics from all the packet engines. The NO setting allows consideration of only the number of connections on the packet engine that received the new connection.

usePortForHashLb

Include the port number of the service when creating a hash for hash based load balancing methods. With the NO setting, only the IP address of the service is considered when creating a hash.

preferDirectRoute

Perform route lookup for traffic received by the NetScaler appliance, and forward the traffic according to configured routes. Do not set this parameter if you want a wildcard virtual server to direct packets received by the appliance to an intermediary device, such as a firewall, even if their destination is directly connected to the appliance. Route lookup is performed after the packets have been processed and returned by the intermediary device.

startupRRFactor

Used to change the factor of service hits after which vserver will come out of slowstart phase.

monitorSkipMaxClient

When a monitor initiates a connection to a service, do not check to determine whether the number of connections to the service has reached the limit specified by the service's Max Clients setting. Enables monitoring to continue even if the service has reached its connection limit.

monitorConnectionClose

Close monitoring connections by sending the service a connection termination message with the specified bit set.

vServerSpecificMac

Allow a MAC-mode virtual server to accept traffic returned by an intermediary device, such as a firewall, to which the traffic was previously forwarded by another MAC-mode virtual server. The second virtual server can then distribute that traffic across the destination server farm. Also useful when load balancing Branch Repeater appliances.

Note: The second virtual server can also send the traffic to another set of intermediary devices, such as another set of firewalls. If necessary, you can configure multiple MAC-mode virtual servers to pass traffic successively through multiple sets of intermediary devices.

sessionsThreshold

This option is used to get the upper-limit on the number of persistent sessions set by the administrator for this system

Example

```
show lb parameter
```

lb persistentSessions

Sep 22, 2015

The following operations can be performed on "lb persistentSessions":

show | clear

show lb persistentSessions

Get all vserver persistent sessions

Synopsys

show lb persistentSessions [<vServer>]

Arguments

vServer

The name of the virtual server.

summary

fullValues

Outputs

type

Type of Persistence.

typestring

Type of Persistence as String.

srcIP

SOURCE IP.

srcIPv6

SOURCE IPv6 ADDRESS.

destIP

DESTINATION IP.

destIPv6

DESTINATION IPv6 ADDRESS.

flags

IPv6 FLAGS.

destPort

Destination port.

vServerName

Virtual server name.

timeout

Persistent Session timeout.

referenceCount

Reference Count.

sipCallID

SIP CALLID.NOTE: This attribute is deprecated.Replaced by "persistenceParam" field

persistenceParam

Specific persistence information . Callid in case of SIP_CALLID persistence entry , RTSP session id in case of RTSP_SESSIONID persistence entry.

devno

count

stateflag

---

clear lb persistentSessions

Use this command to clear/flush persistent sessions

Synopsys

clear lb persistentSessions [<vServer>] [-persistenceParameter <string>]

Arguments

vServer

The name of the LB vserver whose persistence sessions are to be flushed. If not specified, all persistence sessions will be flushed .

persistenceParameter

The persistence parameter whose persistence sessions are to be flushed.

---

lb route

Sep 22, 2015

The following operations can be performed on "lb route":

add | rm | show

add lb route

Bind the route VIP to the route structure.

Synopsys

add lb route <network> <netmask> <gatewayName>

Arguments

network

The IP address of the network to which the route belongs.

netmask

The netmask to which the route belongs.

gatewayName

The name of the route.

---

rm lb route

Remove the route VIP from the route structure.

Synopsys

rm lb route <network> <netmask>

Arguments

network

The IP address of the network to which the route VIP belongs.

netmask

The netmask of the destination network.

---

show lb route

Display the names of the routes associated to the route structure using the ###add lb route### command.

Synopsys

show lb route [<network> <netmask>]

Arguments

network

The destination network or host.

summary

fullValues

format

level

Outputs

gatewayName

The name of the route.

flags

State of the configured gateway.

devno

count

stateflag

lb route6

Sep 22, 2015

The following operations can be performed on "lb route6":

add | rm | show

add lb route6

Bind the route VIP to the route structure.

Synopsys

add lb route6 <network> [<gatewayName>]

Arguments

network

The destination network.

gatewayName

The name of the route.

---

rm lb route6

Remove the route VIP from the route structure.

Synopsys

rm lb route6 <network>

Arguments

network

The IP address of the network to which the route VIP belongs.

---

show lb route6

Display the names of the routes associated to the route structure using the ###add lb route6### command.

Synopsys

show lb route6 [<network>]

Arguments

network

The destination network or host.

summary

fullValues

format

level

Outputs

gatewayName

The name of the route.

flags

State of the configured gateway.

devno

count

stateflag

lb sipParameters

Sep 22, 2015

The following operations can be performed on "lb sipParameters":

set | unset | show

set lb sipParameters

Modifies the specified global SIP parameters.

Synopsys

set lb sipParameters [-rnatSrcPort <port>] [-rnatDstPort <port>] [-retryDur <integer>] [-addRportVip (ENABLED | DISABLED)] [-sip503RateThreshold <positive_integer>]

Arguments

rnatSrcPort

Port number with which to match the source port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

rnatDstPort

Port number with which to match the destination port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

retryDur

Time, in seconds, for which a client must wait before initiating a connection after receiving a 503 Service Unavailable response from the SIP server. The time value is sent in the "Retry-After" header in the 503 response.

Default value: 120

Minimum value: 1

addRportVip

Add the rport parameter to the VIA headers of SIP requests that virtual servers receive from clients or servers.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sip503RateThreshold

Maximum number of 503 Service Unavailable responses to generate, once every 10 milliseconds, when a SIP virtual server becomes unavailable.

Default value: 100

Example

set sip parameter

---

unset lb sipParameters

Use this command to remove lb sipParameters settings.Refer to the set lb sipParameters command for meanings of the arguments.

Synopsys

unset lb sipParameters [-rnatSrcPort] [-rnatDstPort] [-retryDur] [-addRportVip] [-sip503RateThreshold]

---

show lb sipParameters

Displays the global SIP parameters.

Synopsys

show lb sipParameters

Arguments

format

level

Outputs

rnatSrcPort

Port number with which to match the source port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

rnatDstPort

Port number with which to match the destination port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

retryDur

Time, in seconds, for which a client must wait before initiating a connection after receiving a 503 Service Unavailable response from the SIP server. The time value is sent in the "Retry-After" header in the 503 response.

addRportVip

Add the rport parameter to the VIA headers of SIP requests that virtual servers receive from clients or

servers.

sip503RateThreshold

Maximum number of 503 Service Unavailable responses to generate, once every 10 milliseconds, when a SIP virtual server becomes unavailable.

Example

```
show sip parameter
```

lb vserver

Sep 22, 2015

The following operations can be performed on "lb vserver":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

add lb vserver

Creates a load balancing virtual server.

Synopsys

add lb vserver <name>@ <serviceType> [(<IPAddress>@ <port> [-range <positive_integer>]) | (-IPPattern <ippat> -IPMask <ipmask>)] [-persistenceType <persistenceType>] [-timeout <mins>] [-persistenceBackup (SOURCEIP | NONE)] [-backupPersistenceTimeout <mins>] [-lbMethod <lbMethod> [-hashLength <positive_integer>] [-netmask <netmask>] [-v6netmasklen <positive_integer>] [-dataLength <positive_integer>] [-dataOffset <positive_integer>]] [-cookieName <string>] [-rule <expression>] [-Listenpolicy <expression> [-Listenpriority <positive_integer>]] [-resRule <expression>] [-persistMask <netmask>] [-v6persistmasklen <positive_integer>] [-pq (ON | OFF)] [-sc (ON | OFF)] [-rtspNat (ON | OFF)] [-m <m>] [-tosId <positive_integer>] [-sessionless (ENABLED | DISABLED)] [-state (ENABLED | DISABLED)] [-connfailover <connfailover>] [-redirectURL <URL>] [-cacheable (YES | NO)] [-cltTimeout <secs>] [-soMethod <soMethod>] [-soPersistence (ENABLED | DISABLED)] [-soPersistenceTimeOut <positive_integer>] [-healthThreshold <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-redirectPortRewrite (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-backupVServer <string>] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-insertVserverIPPort <insertVserverIPPort> [<vipHeader>]] [-AuthenticationHost <string>] [-Authentication (ON | OFF)] [-authn401 (ON | OFF)] [-authnVsName <string>] [-push (ENABLED | DISABLED)] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients (YES | NO)] [-tcpProfileName <string>] [-httpProfileName <string>] [-dbProfileName <string>] [-comment <string>] [-l2Conn (ON | OFF)] [-mssqlServerVersion <mssqlServerVersion>] [-mysqlProtocolVersion <positive_integer>] [-mysqlServerVersion <string>] [-mysqlCharacterSet <positive_integer>] [-mysqlServerCapabilities <positive_integer>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-icmpVsrResponse (PASSIVE | ACTIVE)] [-newServiceRequest <positive_integer> [<newServiceRequestUnit>]] [-newServiceRequestIncrementInterval <positive_integer>] [-minAutoscaleMembers <positive_integer>] [-maxAutoscaleMembers <positive_integer>] [-persistAVPno <positive_integer> ...] [-skippersistency <skippersistency>] [-td <positive_integer>] [-authnProfile <string>] [-macmodeRetainvlan (ENABLED | DISABLED)] [-dbsLb (ENABLED | DISABLED)] [-dns64 (ENABLED | DISABLED)] [-bypassAAAA (YES | NO)] [-RecursionAvailable (YES | NO)]

Arguments

name

Name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceType

Protocol used by the service (also called the service type).

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, DNS, DHCPRA, ANY, SIP_UDP, DNS_TCP, RTSP, PUSH, SSL_PUSH, RADIUS, RDP, MYSQL, MSSQL, DIAMETER, SSL_DIAMETER, TFTP

IPAddress

IPv4 or IPv6 address to assign to the virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if virtual servers vs1 and vs2 have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

port

Port number for the virtual server.

range

Number of IP addresses that the appliance must generate and assign to the virtual server. The virtual server then functions as a network virtual server, accepting traffic on any of the generated IP addresses. The IP addresses are generated automatically, as follows:

* For a range of n, the last octet of the address specified by the IP Address parameter increments n-1 times.

* If the last octet exceeds 255, it rolls over to 0 and the third octet increments by 1.

Note: The Range parameter assigns multiple IP addresses to one virtual server. To generate an array of virtual servers, each of which owns only one IP address, use brackets in the IP Address and Name parameters to specify the range. For example:

add lb vserver my_vserver[1-3] HTTP 192.0.2.[1-3] 80

Default value: 1

Minimum value: 1

Maximum value: 254

persistenceType

Type of persistence for the virtual server. Available settings function as follows:

* SOURCEIP - Connections from the same client IP address belong to the same persistence session.

* COOKIEINSERT - Connections that have the same HTTP Cookie, inserted by a Set-Cookie directive from a server, belong to the same persistence session.

* SSLSESSION - Connections that have the same SSL Session ID belong to the same persistence session.

* CUSTOMSERVERID - Connections with the same server ID form part of the same session. For this persistence type, set the Server ID (CustomServerID) parameter for each service and configure the Rule parameter to identify the server ID in a request.

* RULE - All connections that match a user defined rule belong to the same persistence session.

* URLPASSIVE - Requests that have the same server ID in the URL query belong to the same persistence session. The server ID is the hexadecimal representation of the IP address and port of the service to which the request must be forwarded. This persistence type requires a rule to identify the server ID in the request.

* DESTIP - Connections to the same destination IP address belong to the same persistence session.

* SRCIPDESTIP - Connections that have the same source IP address and destination IP address belong to the same persistence session.

* CALLID - Connections that have the same CALL-ID SIP header belong to the same persistence session.

* RTSPSID - Connections that have the same RTSP Session ID belong to the same persistence session.

Possible values: SOURCEIP, COOKIEINSERT, SSLSESSION, RULE, URLPASSIVE, CUSTOMSERVERID, DESTIP, SRCIPDESTIP, CALLID, RTSPSID, DIAMETER, NONE

timeout

Time period for which a persistence session is in effect.

Default value: 2

Maximum value: 1440

persistenceBackup

Backup persistence type for the virtual server. Becomes operational if the primary persistence mechanism fails.

Possible values: SOURCEIP, NONE

backupPersistenceTimeout

Time period for which backup persistence is in effect.

Default value: 2

Minimum value: 2

Maximum value: 1440

lbMethod

Load balancing method. The available settings function as follows:

* ROUNDROBIN - Distribute requests in rotation, regardless of the load. Weights can be assigned to services to enforce weighted round robin distribution.

* LEASTCONNECTION (default) - Select the service with the fewest connections.

* LEASTRESPONSETIME - Select the service with the lowest average response time.

* LEASTBANDWIDTH - Select the service currently handling the least traffic.

* LEASTPACKETS - Select the service currently serving the lowest number of packets per second.

* CUSTOMLOAD - Base service selection on the SNMP metrics obtained by custom load monitors.

* LRTM - Select the service with the lowest response time. Response times are learned through monitoring probes. This method also takes the number of active connections into account.

Also available are a number of hashing methods, in which the appliance extracts a predetermined portion of the request, creates a hash of the portion, and then checks whether any previous requests had the same hash value. If it finds a match, it forwards the request to the service that served those previous requests. Following are the hashing methods:

* URLHASH - Create a hash of the request URL (or part of the URL).

* DOMAINHASH - Create a hash of the domain name in the request (or part of the domain name). The domain name is taken from either the URL or the Host header. If the domain name appears in both locations, the URL is preferred. If the request does not contain a domain name, the load balancing method defaults to LEASTCONNECTION.

* DESTINATIONIPHASH - Create a hash of the destination IP address in the IP header.

* SOURCEIPHASH - Create a hash of the source IP address in the IP header.

* TOKEN - Extract a token from the request, create a hash of the token, and then select the service to which any previous requests with the same token hash value were sent.

* SRCIPDESTIPHASH - Create a hash of the string obtained by concatenating the source IP address and destination IP address in the IP header.

* SRCIPSRCPORTHASH - Create a hash of the source IP address and source port in the IP header.

* CALLIDHASH - Create a hash of the SIP Call-ID header.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, URLHASH, DOMAINHASH, DESTINATIONIPHASH, SOURCEIPHASH, SRCIPDESTIPHASH, LEASTBANDWIDTH, LEASTPACKETS, TOKEN, SRCIPSRCPORTHASH, LRTM, CALLIDHASH, CUSTOMLOAD, LEASTREQUEST

Default value: PEMGMT_LB_LEASTCONNS

cookieName

Use this parameter to specify the cookie name for COOKIE peristence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Default value: "none"

Listenpolicy

Default syntax expression identifying traffic accepted by the virtual server. Can be either an expression (for example, CLIENT.IP.DST.IN_SUBNET(192.0.2.0/24) or the name of a named expression. In the above example, the virtual server accepts all requests whose destination IP address is in the 192.0.2.0/24 subnet.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Maximum value: 101

resRule

Default syntax expression specifying which part of a server's response to use for creating rule based persistence sessions (persistence type RULE). Can be either an expression or the name of a named expression.

Example:

HTTP.RES.HEADER("setcookie").VALUE(0).TYPECAST_NVLIST_T('=',';').VALUE("server1").

Default value: "none"

persistMask

Persistence mask for IP based persistence types, for IPv4 virtual servers.

Default value: 0xFFFFFFFF

v6persistmasklen

Persistence mask for IP based persistence types, for IPv6 virtual servers.

Default value: 128

Minimum value: 1

Maximum value: 128

pq

Use priority queuing on the virtual server. based persistence types, for IPv6 virtual servers.

Possible values: ON, OFF

Default value: OFF

sc

Use SureConnect on the virtual server.

Possible values: ON, OFF

Default value: OFF

rtspNat

Use network address translation (NAT) for RTSP data connections.

Possible values: ON, OFF

Default value: OFF

m

Redirection mode for load balancing. Available settings function as follows:

* IP - Before forwarding a request to a server, change the destination IP address to the server's IP address.

* MAC - Before forwarding a request to a server, change the destination MAC address to the server's MAC address. The destination IP address is not changed. MAC-based redirection mode is used mostly in firewall load balancing deployments.

* IPTUNNEL - Perform IP-in-IP encapsulation for client IP packets. In the outer IP headers, set the destination IP address to the IP address of the server and the source IP address to the subnet IP (SNIP). The client IP packets are not modified. Applicable to both IPv4 and IPv6 packets.

* TOS - Encode the virtual server's TOS ID in the TOS field of the IP header.

You can use either the IPTUNNEL or the TOS option to implement Direct Server Return (DSR).

Possible values: IP, MAC, IPTUNNEL, TOS

Default value: NSFWD_IP

tosId

TOS ID of the virtual server. Applicable only when the load balancing redirection mode is set to TOS.

Minimum value: 1

Maximum value: 63

dataLength

Length of the token to be extracted from the data segment of an incoming packet, for use in the token method of load balancing. The length of the token, specified in bytes, must not be greater than 24 KB. Applicable to virtual servers of type TCP.

Minimum value: 1

Maximum value: 100

dataOffset

Offset to be considered when extracting a token from the TCP payload. Applicable to virtual servers, of type TCP, using the token method of load balancing. Must be within the first 24 KB of the TCP payload.

Maximum value: 25400

sessionless

Perform load balancing on a per-packet basis, without establishing sessions. Recommended for load balancing of intrusion detection system (IDS) servers and scenarios involving direct server return (DSR), where session information is unnecessary.

Possible values: ENABLED, DISABLED

Default value: DISABLED

state

State of the load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

connfailover

Mode in which the connection failover feature must operate for the virtual server. After a failover, established TCP connections and UDP packet flows are kept active and resumed on the secondary appliance. Clients remain connected to the same servers. Available settings function as follows:

* STATEFUL - The primary appliance shares state information with the secondary appliance, in real time, resulting in some runtime processing overhead.

* STATELESS - State information is not shared, and the new primary appliance tries to re-create the packet flow on the basis of the information contained in the packets it receives.

* DISABLED - Connection failover does not occur.

Possible values: DISABLED, STATEFUL, STATELESS

Default value: DISABLED

redirectURL

URL to which to redirect traffic if the virtual server becomes unavailable.

WARNING! Make sure that the domain in the URL does not match the domain specified for a content switching policy. If it does, requests are continuously redirected to the unavailable

virtual server.

cacheable

Route cacheable requests to a cache redirection virtual server. The load balancing virtual server can forward requests only to a transparent cache redirection virtual server that has an IP address and port combination of *:80, so such a cache redirection virtual server must be configured on the appliance.

Possible values: YES, NO

Default value: NO

cltTimeout

Idle time, in seconds, after which a client connection is terminated.

Default value: VAL_NOT_SET

Maximum value: 31536000

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the virtual server exceeds the sum of the maximum client (Max Clients) settings for bound services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of bound services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the services that are UP drops below the threshold. For example, if services svc1, svc2, and svc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if svc1 and svc3 or svc2 and svc3 transition to DOWN.

* NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

healthThreshold

Threshold in percent of active services below which vserver state is made down. If this threshold is 0, vserver state will be up even if one bound service is up.

Maximum value: 100

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

Rewrite the port and change the protocol to ensure successful HTTP redirects from services.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

backupVServer

Name of the backup virtual server to which to forward requests if the primary virtual server goes DOWN or reaches its spillover threshold.

disablePrimaryOnDown

If the primary virtual server goes down, do not allow it to return to primary status until manually enabled.

Possible values: ENABLED, DISABLED

Default value: DISABLED

insertVserverIPPort

Insert an HTTP header, whose value is the IP address and port number of the virtual server, before forwarding a request to the server. The format of the header is <vipHeader>: <virtual server IP address>_<port number >, where vipHeader is the name that you specify for the header. If the virtual server has an IPv6 address, the address in the header is enclosed in brackets ([and]) to separate it from the port number. If you have mapped an IPv4 address to a virtual server's IPv6 address, the value of this parameter determines which IP address is inserted in the header, as follows:

* VIPADDR - Insert the IP address of the virtual server in the HTTP header regardless of whether the virtual server has an IPv4 address or an IPv6 address. A mapped IPv4 address, if configured, is ignored.

* V6TOV4MAPPING - Insert the IPv4 address that is mapped to the virtual server's IPv6 address. If a mapped IPv4 address is not configured, insert the IPv6 address.

* OFF - Disable header insertion.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

AuthenticationHost

Fully qualified domain name (FQDN) of the authentication virtual server to which the user must be redirected for authentication. Make sure that the Authentication parameter is set to ENABLED.

Authentication

Enable or disable user authentication.

Possible values: ON, OFF

Default value: OFF

authn401

Enable or disable user authentication with HTTP 401 responses.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of an authentication virtual server with which to authenticate users.

push

Process traffic with the push virtual server that is bound to this load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the load balancing virtual server that you are configuring.

pushLabel

Expression for extracting a label from the server's response. Can be either an expression or the name of a named expression.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

tcpProfileName

Name of the TCP profile whose settings are to be applied to the virtual server.

httpProfileName

Name of the HTTP profile whose settings are to be applied to the virtual server.

dbProfileName

Name of the DB profile whose settings are to be applied to the virtual server.

comment

Any comments that you might want to associate with the virtual server.

l2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to co-exist on the NetScaler appliance.

Possible values: ON, OFF

mssqlServerVersion

For a load balancing virtual server of type MSSQL, the Microsoft SQL Server version. Set this parameter if you expect some clients to run a version different from the version of the database. This setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: TDS_PROT_2008B

mysqlProtocolVersion

MySQL protocol version that the virtual server advertises to clients.

Default value: NSA_MYSQL_PROTOCOL_VER_DEFAULT

mysqlServerVersion

MySQL server version string that the virtual server advertises to clients.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

Character set that the virtual server advertises to clients.

Default value: NSA_MYSQL_CHAR_SET_DEFAULT

mysqlServerCapabilities

Server capabilities that the virtual server advertises to clients.

Default value: NSA_MYSQL_SVR_CAPABILITIES_DEFAULT

appflowLog

Apply AppFlow logging to the virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Name of the network profile to associate with the virtual server. If you set this parameter, the virtual server uses only the IP addresses in the network profile as source IP addresses when initiating connections with servers.

icmpVsrResponse

How the NetScaler appliance responds to ping requests received for an IP address that is common to one or more virtual servers. Available settings function as follows:

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always responds to the ping requests.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance responds to the ping requests if at least one of the virtual servers is UP. Otherwise, the appliance does not respond.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance responds if at least one virtual server with the ACTIVE setting is UP. Otherwise, the appliance does not respond.

Note: This parameter is available at the virtual server level. A similar parameter, ICMP Response, is available at the IP address level, for IPv4 addresses of type VIP. To set that parameter, use the add ip command in the CLI or the Create IP dialog box in the GUI.

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

newServiceRequest

Number of requests, or percentage of the load on existing services, by which to increase the load on a new service at each interval in slow-start mode. A value of 0 disables slow start for new services, and any services still ramping up. They immediately receive their full share of the load. If you use the CLI, be sure to set the newServiceRequestUnit parameter to specify the unit as number of requests or percentage of load.

newServiceRequestIncrementInterval

Interval, in seconds, between successive increments in the load on a new service or a service whose state has just changed from DOWN to UP. A value of 0 (zero) specifies manual slow start.

Maximum value: 3600

minAutoscaleMembers

Minimum number of members expected to be present when vserver is used in Autoscale.

Maximum value: 5000

maxAutoscaleMembers

Maximum number of members expected to be present when vserver is used in Autoscale.

Maximum value: 5000

persistAVPno

Persist AVP number for Diameter Persistency.

In case this AVP is not defined in Base RFC 3588 and it is nested inside a Grouped AVP,

define a sequence of AVP numbers (max 3) in order of parent to child. So say persist AVP number X

is nested inside AVP Y which is nested in Z, then define the list as Z Y X

Minimum value: 1

skippersistency

This argument decides the behavior incase the service which is selected from an existing persistence session has reached threshold.

Possible values: Bypass, ReLb, None

Default value: NS_DONT_SKIPPERSIST

td

Traffic Domain ID

Maximum value: 4094

authnProfile

Name of the authentication profile to be used when authentication is turned on.

macmodeRetainvlan

This option is used to retain vlan information of incoming packet when macmode is enabled

Possible values: ENABLED, DISABLED

Default value: DISABLED

dbsLb

For enabling database specific load-balacing..

Possible values: ENABLED, DISABLED

Default value: DISABLED

dns64

This argument is for enabling/disabling the dns64 on lbvserver

Possible values: ENABLED, DISABLED

bypassAAAA

If this option is enabled while resolving DNS64 query AAAA queries are not sent to back end dns server

Possible values: YES, NO

Default value: NO

RecursionAvailable

When set to YES, this option causes the DNS replies from this vserver to have the RA bit turned on. Typically one would set this option to YES, when the vserver is load balancing a set of DNS servers thatsupport recursive queries.

Possible values: YES, NO

Default value: NO

Example

add lb vserver http_vsvr http 10.102.1.10 80 To add multiple vservers at once use the following command: add lb vs http_vsvr[1-4] http 10.102.27.[115-118] 80 This com

---

rm lb vserver

Removes a virtual server from the NetScaler appliance.

Synopsys

rm lb vserver <name>@ ...

Arguments

name

Name of the virtual server.

Example

rm vserver lb_vip To remove multiple vservers use the following command: rm vserver lb_vip[1-3]

---

set lb vserver

Modifies the specified parameters of a load balancing virtual server.

Synopsys

set lb vserver <name>@ [-IPAddress <ip_addr|ipv6_addr|*>@] [-IPPattern <ippat>] [-IPMask <ipmask>] [-weight <positive_integer> <serviceName>@] [-persistenceType <persistenceType>] [-timeout <mins>] [-persistenceBackup (SOURCEIP | NONE)] [-backupPersistenceTimeout <mins>] [-lbMethod <lbMethod> [-hashLength <positive_integer>] [-netmask <netmask>] [-v6netmasklen <positive_integer>]] [-rule <expression>] [-cookieName <string>] [-resRule <expression>] [-persistMask <netmask>] [-v6persistmasklen <positive_integer>] [-pq (ON | OFF)] [-sc (ON | OFF)] [-rtspNat (ON | OFF)] [-m <m>] [-tosId <positive_integer>] [-dataLength <positive_integer>] [-dataOffset <positive_integer>] [-sessionless (ENABLED | DISABLED)] [-connfailover <connfailover>] [-backupVServer <string>] [-redirectURL <URL>] [-cacheable (YES | NO)] [-cltTimeout <secs>] [-soMethod <soMethod>] [-soThreshold <positive_integer>] [-soPersistence (ENABLED | DISABLED)] [-soPersistenceTimeOut <positive_integer>] [-healthThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-redirectPortRewrite (ENABLED | DISABLED)] [-downStateFlush (ENABLED | DISABLED)] [-insertVserverIPPort <insertVserverIPPort> [<vipHeader>]] [-disablePrimaryOnDown (ENABLED | DISABLED)] [-AuthenticationHost <string>] [-Authentication (ON | OFF)] [-authn401 (ON | OFF)] [-authnVsName <string>] [-push (ENABLED | DISABLED)] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients (YES | NO)] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-dbProfileName <string>] [-comment <string>] [-l2Conn (ON | OFF)] [-mssqlServerVersion <mssqlServerVersion>] [-mysqlProtocolVersion <positive_integer>] [-mysqlServerVersion <string>] [-mysqlCharacterSet <positive_integer>] [-mysqlServerCapabilities <positive_integer>] [-appflowLog (ENABLED | DISABLED)] [-netProfile <string>] [-icmpVsrResponse (PASSIVE | ACTIVE)] [-newServiceRequest <positive_integer>] [<newServiceRequestUnit>] [-newServiceRequestIncrementInterval <positive_integer>] [-minAutoscaleMembers <positive_integer>] [-maxAutoscaleMembers <positive_integer>] [-persistAVPno <positive_integer> ...] [-skippersistency <skippersistency>] [-authnProfile <string>] [-macmodeRetainvlan (ENABLED | DISABLED)] [-dbsLb (ENABLED | DISABLED)] [-dns64 (ENABLED | DISABLED)] [-bypassAAAA (YES | NO)] [-RecursionAvailable (YES | NO)]

Arguments

name

Name of the virtual server.

IPAddress

IPv4 or IPv6 address to assign to the virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if virtual servers vs1 and vs2 have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

IPMask

IP mask, in dotted decimal notation, for the IP Pattern parameter. Can have leading or trailing non-zero octets (for example, 255.255.240.0 or 0.0.255.255). Accordingly, the mask specifies whether the first n bits or the last n bits of the destination IP address in a client request are to be matched with the corresponding bits in the IP pattern. The former is called a forward mask. The latter is called a reverse mask.

weight

Weight to assign to the specified service.

Minimum value: 1

Maximum value: 100

persistenceType

Type of persistence for the virtual server. Available settings function as follows:

* SOURCEIP - Connections from the same client IP address belong to the same persistence session.

* COOKIEINSERT - Connections that have the same HTTP Cookie, inserted by a Set-Cookie directive from a server, belong to the same persistence session.

* SSLSESSION - Connections that have the same SSL Session ID belong to the same persistence session.

* CUSTOMSERVERID - Connections with the same server ID form part of the same session. For this persistence type, set the Server ID (CustomServerID) parameter for each service and configure the Rule parameter to identify the server ID in a request.

* RULE - All connections that match a user defined rule belong to the same persistence session.

* URLPASSIVE - Requests that have the same server ID in the URL query belong to the same persistence session. The server ID is the hexadecimal representation of the IP address and port of the service to which the request must be forwarded. This persistence type requires a rule to identify the server ID in the request.

* DESTIP - Connections to the same destination IP address belong to the same persistence session.

* SRCIPDESTIP - Connections that have the same source IP address and destination IP address belong to the same persistence session.

* CALLID - Connections that have the same CALL-ID SIP header belong to the same persistence session.

* RTSPSID - Connections that have the same RTSP Session ID belong to the same persistence session.

Possible values: SOURCEIP, COOKIEINSERT, SSLSESSION, RULE, URLPASSIVE, CUSTOMSERVERID, DESTIP, SRCIPDESTIP, CALLID, RTSPSID, DIAMETER, NONE

timeout

Time period for which a persistence session is in effect.

Default value: 2

Maximum value: 1440

persistenceBackup

Backup persistence type for the virtual server. Becomes operational if the primary persistence mechanism fails.

Possible values: SOURCEIP, NONE

backupPersistenceTimeout

Time period for which backup persistence is in effect.

Default value: 2

Minimum value: 2

Maximum value: 1440

lbMethod

Load balancing method. The available settings function as follows:

* ROUNDROBIN - Distribute requests in rotation, regardless of the load. Weights can be assigned to services to enforce weighted round robin distribution.

* LEASTCONNECTION (default) - Select the service with the fewest connections.

* LEASTRESPONSETIME - Select the service with the lowest average response time.

* LEASTBANDWIDTH - Select the service currently handling the least traffic.

* LEASTPACKETS - Select the service currently serving the lowest number of packets per second.

* CUSTOMLOAD - Base service selection on the SNMP metrics obtained by custom load monitors.

* LRTM - Select the service with the lowest response time. Response times are learned through monitoring probes. This method also takes the number of active connections into account.

Also available are a number of hashing methods, in which the appliance extracts a predetermined portion of the request, creates a hash of the portion, and then checks whether any previous requests had the same hash value. If it finds a match, it forwards the request to the service that served those previous requests. Following are the hashing methods:

* URLHASH - Create a hash of the request URL (or part of the URL).

* DOMAINHASH - Create a hash of the domain name in the request (or part of the domain name). The domain name is taken from either the URL or the Host header. If the domain name appears in both locations, the URL is preferred. If the request does not contain a domain name, the load balancing method defaults to LEASTCONNECTION.

* DESTINATIONIPHASH - Create a hash of the destination IP address in the IP header.

* SOURCEIPHASH - Create a hash of the source IP address in the IP header.

* TOKEN - Extract a token from the request, create a hash of the token, and then select the service to which any previous requests with the same token hash value were sent.

* SRCIPDESTIPHASH - Create a hash of the string obtained by concatenating the source IP address and destination IP address in the IP header.

* SRCIPSRCPORTHASH - Create a hash of the source IP address and source port in the IP header.

* CALLIDHASH - Create a hash of the SIP Call-ID header.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, URLHASH, DOMAINHASH, DESTINATIONIPHASH, SOURCEIPHASH, SRCIPDESTIPHASH, LEASTBANDWIDTH, LEASTPACKETS, TOKEN, SRCIPSRCPORTHASH, LRTM, CALLIDHASH, CUSTOMLOAD, LEASTREQUEST

Default value: PEMGMT_LB_LEASTCONNS

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "'<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Default value: "none"

cookieName

Use this parameter to specify the cookie name for COOKIE peristence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

resRule

Default syntax expression specifying which part of a server's response to use for creating rule based persistence sessions (persistence type RULE). Can be either an expression or the name of a named expression.

Example:

HTTP.RES.HEADER("setcookie").VALUE(0).TYPECAST_NVLIST_T('=',';').VALUE("server1").

Default value: "none"

persistMask

Persistence mask for IP based persistence types, for IPv4 virtual servers.

Default value: 0xFFFFFFFF

v6persistmasklen

Persistence mask for IP based persistence types, for IPv6 virtual servers.

Default value: 128

Minimum value: 1

Maximum value: 128

pq

Use priority queuing on the virtual server. based persistence types, for IPv6 virtual servers.

Possible values: ON, OFF

Default value: OFF

sc

Use SureConnect on the virtual server.

Possible values: ON, OFF

Default value: OFF

rtspNat

Use network address translation (NAT) for RTSP data connections.

Possible values: ON, OFF

Default value: OFF

m

Redirection mode for load balancing. Available settings function as follows:

* IP - Before forwarding a request to a server, change the destination IP address to the server's IP address.

* MAC - Before forwarding a request to a server, change the destination MAC address to the server's MAC address. The destination IP address is not changed. MAC-based redirection mode is used mostly in firewall load balancing deployments.

* IPTUNNEL - Perform IP-in-IP encapsulation for client IP packets. In the outer IP headers, set the destination IP address to the IP address of the server and the source IP address to the subnet IP (SNIP). The client IP packets are not modified. Applicable to both IPv4 and IPv6 packets.

* TOS - Encode the virtual server's TOS ID in the TOS field of the IP header.

You can use either the IPTUNNEL or the TOS option to implement Direct Server Return (DSR).

Possible values: IP, MAC, IPTUNNEL, TOS

Default value: NSFWD_IP

tosId

TOS ID of the virtual server. Applicable only when the load balancing redirection mode is set to TOS.

Minimum value: 1

Maximum value: 63

dataLength

Length of the token to be extracted from the data segment of an incoming packet, for use in the token method of load balancing. The length of the token, specified in bytes, must not be greater than 24 KB. Applicable to virtual servers of type TCP.

Minimum value: 1

Maximum value: 100

dataOffset

Offset to be considered when extracting a token from the TCP payload. Applicable to virtual servers, of type TCP, using the token method of load balancing. Must be within the first 24 KB of the TCP payload.

Maximum value: 25400

sessionless

Perform load balancing on a per-packet basis, without establishing sessions. Recommended for load balancing of intrusion detection system (IDS) servers and scenarios involving direct server return (DSR), where session information is unnecessary.

Possible values: ENABLED, DISABLED

Default value: DISABLED

connfailover

Mode in which the connection failover feature must operate for the virtual server. After a failover, established TCP connections and UDP packet flows are kept active and resumed on the secondary appliance. Clients remain connected to the same servers. Available settings function as follows:

* STATEFUL - The primary appliance shares state information with the secondary appliance, in real time, resulting in some runtime processing overhead.

* STATELESS - State information is not shared, and the new primary appliance tries to re-create the packet flow on the basis of the information contained in the packets it receives.

* DISABLED - Connection failover does not occur.

Possible values: DISABLED, STATEFUL, STATELESS

Default value: DISABLED

backupVServer

Name of the backup virtual server to which to forward requests if the primary virtual server goes DOWN or reaches its spillover threshold.

redirectURL

URL to which to redirect traffic if the virtual server becomes unavailable.

WARNING! Make sure that the domain in the URL does not match the domain specified for a content switching policy. If it does, requests are continuously redirected to the unavailable virtual server.

cacheable

Route cacheable requests to a cache redirection virtual server. The load balancing virtual server can forward requests only to a transparent cache redirection virtual server that has an IP address and port combination of *:80, so such a cache redirection virtual server must be configured on the appliance.

Possible values: YES, NO

Default value: NO

cltTimeout

Idle time, in seconds, after which a client connection is terminated.

Default value: VAL_NOT_SET

Maximum value: 31536000

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the virtual server exceeds the sum of the maximum client (Max Clients) settings for bound services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of bound services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the services that are UP drops below the threshold. For example, if services svc1, svc2, and svc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if svc1 and svc3 or svc2 and svc3 transition to DOWN.

* NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

healthThreshold

Threshold in percent of active services below which vserver state is made down. If this threshold is 0, vserver state will be up even if one bound service is up.

Maximum value: 100

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

Rewrite the port and change the protocol to ensure successful HTTP redirects from services.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

insertVserverIPPort

Insert an HTTP header, whose value is the IP address and port number of the virtual server, before forwarding a request to the server. The format of the header is <vipHeader>: <virtual server IP address>_<port number >, where vipHeader is the name that you specify for the header. If the virtual server has an IPv6 address, the address in the header is enclosed in brackets ([and]) to separate it from the port number. If you have mapped an IPv4 address to a virtual server's IPv6 address, the value of this parameter determines which IP address is inserted in the header, as follows:

* VIPADDR - Insert the IP address of the virtual server in the HTTP header regardless of whether the virtual server has an IPv4 address or an IPv6 address. A mapped IPv4 address, if configured, is ignored.

* V6TOV4MAPPING - Insert the IPv4 address that is mapped to the virtual server's IPv6 address. If a mapped IPv4 address is not configured, insert the IPv6 address.

* OFF - Disable header insertion.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

disablePrimaryOnDown

If the primary virtual server goes down, do not allow it to return to primary status until manually enabled.

Possible values: ENABLED, DISABLED

Default value: DISABLED

AuthenticationHost

Fully qualified domain name (FQDN) of the authentication virtual server to which the user must be redirected for authentication. Make sure that the Authentication parameter is set to ENABLED.

Authentication

Enable or disable user authentication.

Possible values: ON, OFF

Default value: OFF

authn401

Enable or disable user authentication with HTTP 401 responses.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of an authentication virtual server with which to authenticate users.

push

Process traffic with the push virtual server that is bound to this load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the load balancing virtual server that you are configuring.

pushLabel

Expression for extracting a label from the server's response. Can be either an expression or the name of a named expression.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

Listenpolicy

Default syntax expression identifying traffic accepted by the virtual server. Can be either an expression (for example, CLIENT.IP.DST.IN_SUBNET(192.0.2.0/24) or the name of a named expression. In the above example, the virtual server accepts all requests whose destination IP address is in the 192.0.2.0/24 subnet.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Maximum value: 101

tcpProfileName

Name of the TCP profile whose settings are to be applied to the virtual server.

httpProfileName

Name of the HTTP profile whose settings are to be applied to the virtual server.

dbProfileName

Name of the DB profile whose settings are to be applied to the virtual server.

comment

Any comments that you might want to associate with the virtual server.

l2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to co-exist on the NetScaler appliance.

Possible values: ON, OFF

mssqlServerVersion

For a load balancing virtual server of type MSSQL, the Microsoft SQL Server version. Set this parameter if you expect some clients to run a version different from the version of the database. This setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: TDS_PROT_2008B

mysqlProtocolVersion

MySQL protocol version that the virtual server advertises to clients.

Default value: NSA_MYSQL_PROTOCOL_VER_DEFAULT

mysqlServerVersion

MySQL server version string that the virtual server advertises to clients.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

Character set that the virtual server advertises to clients.

Default value: NSA_MYSQL_CHAR_SET_DEFAULT

mysqlServerCapabilities

Server capabilities that the virtual server advertises to clients.

Default value: NSA_MYSQL_SVR_CAPABILITIES_DEFAULT

appflowLog

Apply AppFlow logging to the virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Name of the network profile to associate with the virtual server. If you set this parameter, the virtual server uses only the IP addresses in the network profile as source IP addresses when initiating connections with servers.

icmpVsrResponse

How the NetScaler appliance responds to ping requests received for an IP address that is common to one or more virtual servers. Available settings function as follows:

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always responds to the ping requests.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance responds to the ping requests if at least one of the virtual servers is UP. Otherwise, the appliance does not respond.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance responds if at least one virtual server with the ACTIVE setting is UP. Otherwise, the appliance does not respond.

Note: This parameter is available at the virtual server level. A similar parameter, ICMP Response, is available at the IP address level, for IPv4 addresses of type VIP. To set that parameter, use the add ip command in the CLI or the Create IP dialog box in the GUI.

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

newServiceRequest

Number of requests, or percentage of the load on existing services, by which to increase the load on a new service at each interval in slow-start mode. A value of 0 disables slow start for new services, and any services still ramping up. They immediately receive their full share of the load. If you use the CLI, be sure to set the newServiceRequestUnit parameter to specify the unit as number of requests or percentage of load.

newServiceRequestIncrementInterval

Interval, in seconds, between successive increments in the load on a new service or a service whose state has just changed from DOWN to UP. A value of 0 (zero) specifies manual slow start.

Maximum value: 3600

minAutoscaleMembers

Minimum number of members expected to be present when vserver is used in Autoscale.

Maximum value: 5000

maxAutoscaleMembers

Maximum number of members expected to be present when vserver is used in Autoscale.

Maximum value: 5000

persistAVPno

Persist AVP number for Diameter Persistency.

In case this AVP is not defined in Base RFC 3588 and it is nested inside a Grouped AVP,

define a sequence of AVP numbers (max 3) in order of parent to child. So say persist AVP number X

is nested inside AVP Y which is nested in Z, then define the list as Z Y X

Minimum value: 1

skippersistency

This argument decides the behavior incase the service which is selected from an existing persistence session has reached threshold.

Possible values: Bypass, ReLb, None

Default value: NS_DONT_SKIPPERSIST

authnProfile

Name of the authentication profile to be used when authentication is turned on.

macmodeRetainvlan

This option is used to retain vlan information of incoming packet when macmode is enabled

Possible values: ENABLED, DISABLED

Default value: DISABLED

dbsLb

For enabling database specific load-balacing..

Possible values: ENABLED, DISABLED

Default value: DISABLED

dns64

This argument is for enabling/disabling the dns64 on lbvserver

Possible values: ENABLED, DISABLED

bypassAAAA

If this option is enabled while resolving DNS64 query AAAA queries are not sent to back end dns server

Possible values: YES, NO

Default value: NO

RecursionAvailable

When set to YES, this option causes the DNS replies from this vserver to have the RA bit turned on. Typically one would set this option to YES, when the vserver is load balancing a set of DNS servers thatsupport recursive queries.

Possible values: YES, NO

Default value: NO

Example

set lb vserver http_vip -lbmethod LEASTRESPONSETIME To set the load balancing method for multiple vserver use the following command: set lb vserver http_vip[1-3] -

---

unset lb vserver

Removes the specified parameter settings from the virtual server..Refer to the set lb vserver command for meanings of the arguments.

Synopsys

unset lb vserver <name>@ [-backupVServer] [-cltTimeout] [-redirectURL] [-authn401] [-Authentication] [-AuthenticationHost] [-authnVsName] [-pushVserver] [-pushLabel] [-tcpProfileName] [-httpProfileName] [-dbProfileName] [-rule] [-l2Conn] [-mysqlProtocolVersion] [-mysqlServerVersion] [-mysqlCharacterSet] [-mysqlServerCapabilities] [-appflowLog] [-netProfile] [-icmpVsrResponse] [-skippersistency] [-minAutoscaleMembers] [-maxAutoscaleMembers] [-authnProfile] [-macmodeRetainvlan] [-dbsLb] [-serviceName] [-persistenceType] [-timeout] [-persistenceBackup] [-backupPersistenceTimeout] [-lbMethod] [-hashLength] [-netmask] [-v6netmasklen] [-cookieName] [-resRule] [-persistMask] [-v6persistmasklen] [-pq] [-sc] [-rtspNat] [-m] [-tosId] [-dataLength] [-dataOffset] [-sessionless] [-connfailover] [-cacheable] [-soMethod] [-soPersistence] [-soPersistenceTimeOut] [-healthThreshold] [-soBackupAction] [-redirectPortRewrite] [-downStateFlush] [-insertVserverIPPort] [-vipHeader] [-disablePrimaryOnDown] [-push] [-pushMultiClients] [-Listenpolicy] [-Listenpriority] [-comment] [-mssqlServerVersion] [-newServiceRequest] [-newServiceRequestUnit] [-newServiceRequestIncrementInterval] [-persistAVPno] [-RecursionAvailable]

Example

unset lb vserver lb_vip -backupVServer To unset the backup virtual server for multiple vservers use the following command: unset lb vserver lb_vip[1-3] -backupVServer

---

bind lb vserver

Binds a service, service group, or policy to a virtual server.

Synopsys

bind lb vserver <name>@ ((<serviceName>@ [-weight <positive_integer>]) | <serviceGroupName>@ | (-policyName <string>@ [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type (REQUEST | RESPONSE)] [-invoke (<labelType> <labelName>)]))

Arguments

name

Name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign

(@), equal sign (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceName

Name of the service.

serviceGroupName

Name of the service group.

policyName

Name of the policy to bind to the virtual server.

Example

bind lb vserver http_vip http_svc To bind a service to multiple vservers use the following command: bind lb vs http_vip[1-3] http_svc To bind multiple services to a vserv

---

unbind lb vserver

Unbinds a service, service group, or policy from a virtual server.

Synopsys

unbind lb vserver <name>@ (<serviceName>@ | <serviceGroupName>@ | (-policyName <string>@ [-type (REQUEST | RESPONSE)])) [-priority <positive_integer>]

Arguments

name

Name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceName

Name of the service.

serviceGroupName

The name of the service group that is unbound.

policyName

Name of the policy to bind to the virtual server.

priority

Priority number of the policy.

Minimum value: 1

Maximum value: 2147483647

Example

unbind lb vserver http_vip http_svc To unbind a service from multiple vservers use the following command: unbind lb vs http_vip[1-3] http_svc To unbind multiple se

---

enable lb vserver

Enables a virtual server.

Synopsys

enable lb vserver <name>@

Arguments

name

Name of the virtual server.

Example

enable vserver lb_vip To enable multiple vservers at once use the following command: enable vserver lb_vip[1-3]

---

disable lb vserver

Disables a virtual server.

Synopsys

disable lb vserver <name>@

Arguments

name

Name of the virtual server.

Example

disable vserver lb_vip To disable multiple vservers at once use the following command: disable vserver lb_vip[1-3]

---

show lb vserver

Displays the statistical data collected for a load balancing virtual server.

Synopsys

show lb vserver [<name>] show lb vserver stats - alias for 'stat lb vserver'

Arguments

name

Name of the virtual server. If no name is provided, statistical data of all configured virtual servers is displayed.

summary

fullValues

format

level

Outputs

insertVserverIPPort

The virtual IP and port header insertion option for the vserver.

vipHeader

Name for the inserted header. The default name is vip-header.

value

SSL status.

stateflag

appfwPolicyFlag

IPAddress

IPv4 or IPv6 address to assign to the virtual server.

IPPattern

The IP pattern of the virtual server.

IPMask

The IP address mask of the virtual server.

Listenpolicy

The string is listenpolicy configured for lb vserver

Listenpriority

This parameter is the priority for listen policy of LB Vserver.

IPMapping

The permanent mapping for the V6 Address

port

Port number for the virtual server.

range

Number of IP addresses that the appliance must generate and assign to the virtual server. The virtual server then functions as a network virtual server, accepting traffic on any of the generated IP addresses. The IP addresses are generated automatically, as follows:

* For a range of n, the last octet of the address specified by the IP Address parameter increments n-1 times.

* If the last octet exceeds 255, it rolls over to 0 and the third octet increments by 1.

Note: The Range parameter assigns multiple IP addresses to one virtual server. To generate an array of virtual servers, each of which owns only one IP address, use brackets in the IP Address and Name parameters to specify the range. For example:

add lb vserver my_vserver[1-3] HTTP 192.0.2.[1-3] 80

serviceType

Protocol used by the service (also called the service type).

ngname

Nodegroup name to which this lbvsever belongs to

type

The bindpoint to which the policy is bound

state

State of the load balancing virtual server.

effectiveState

Effective state of the LB vserver , based on the state of backup vservers.

status

Current status of the lb vserver. During the initial phase if the configured lb method is not round robin , the vserver will adopt round robin to distribute traffic for a predefined number of requests.

lbrrreason

Reason why a vserver is in RR. The following are the reasons:

1 - MEP is DOWN (GSLB)

2 - LB method has changed

3 - Bound service's state changed to UP

4 - A new service is bound

5 - Startup RR factor has changed

6 - LB feature is enabled

7 - Load monitor is not active on a service

8 - Vserver is Enabled

9 - SSL feature is Enabled

10 - All bound services have reached threshold. Using effective state to load balance (GSLB)

11 - Primary state of bound services are not UP. Using effective state to load balance (GSLB)

12 - No LB decision can be made as all bound services have either reached threshold or are not UP (GSLB)

13 - All load monitors are active

cacheType

Cache type.

redirect

Cache redirect type.

precedence

Precedence.

redirectURL

The redirect URL.

Authentication

Authentication.

authn401

HTTP 401 response based authentication.

authnVsName

Name of an authentication virtual server with which to authenticate users.

homePage

Home page.

dnsVserverName

DNS vserver name.

domain

Domain.

policyName

Name of the policy bound to the LB vserver.

serviceName

Service to bind to the virtual server.

serviceGroupName

The service group name bound to the selected load balancing virtual server.

weight

Weight to assign to the specified service.

dynamicWeight

Dynamic weight

cacheVserver

Cache virtual server.

backupVServer

Name of the backup virtual server to which to forward requests if the primary virtual server goes DOWN or reaches its spillover threshold.

priority

Priority.

cltTimeout

The client timeout in seconds.

soMethod

The spillover method to be in effect.

soPersistence

State of spillover persistence.

soPersistenceTimeOut

The maximum time persistence is in effect for a specific client on a spillover vserver.

healthThreshold

Threshold in percent of active services below which vserver state is made down.

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

lbMethod

The load balancing method to be in effect

hashLength

The hash length.

dataOffset

The data offset length for TOKEN load balancing method.

health

Health of vserver based on percentage of weights of active svcs/all svcs. This does not consider administratively disabled svcs

dataLength

The data length for TOKEN load balancing method.

netmask

The netmask of the destination network.

v6netmasklen

The netmask of the destination network.

rule

Rule type.

resRule

Use this parameter to specify the expression to be used in response for RULE persistence type.

The string is an in-line expression with a maximum of 1499 characters.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

ruleType

Rule type.

groupName

LB group to which the lb vserver is to be bound.

m

The LB mode.

tosId

TOS ID

persistenceType

The persistence type for the specified virtual server

timeout

The maximum time persistence is in effect for a specific client.

cookieDomain

Domain name to be used in the set cookie header in case of cookie persistence.

persistMask

The persistence mask for v4 traffic

v6persistmasklen

The persistence mask for v6 traffic.

persistenceBackup

The maximum time backup persistence is in effect for a specific client.

backupPersistenceTimeout

Time period for which backup persistence is in effect.

cacheable

The state of caching.

pq

The state of priority queuing on the specified virtual server.

sc

The state of SureConnect the specified virtual server.

rtspNat

Use network address translation (NAT) for RTSP data connections.

sessionless

To enable sessionless load balancing, enable this option

map

Map.

connfailover

The connection failover mode of the virtual server

redirectPortRewrite

Rewrite the port and change the protocol to ensure successful HTTP redirects from services.

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

gt2GB

Allow for greater than 2 GB transactions on this vserver.

consolidatedLConn

Use consolidated stats for LeastConnection.

consolidatedLConnGbl

Fetches Global setting.

thresholdValue

Tells whether threshold exceeded for this service participating in CUSTOMLB

invoke

Invoke policies bound to a virtual server or policy label.

labelType

The invocation type.

labelName

Name of the label invoked.

cookieIpPort

Encryped Ip address and port of the service that is inserted into the set-cookie http header

cookieName

Use this parameter to specify the cookie name for COOKIE peristence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

vserverId

Vserver Id

version

Cookie version

totalServices

Total number of services bound to the vserver.

activeServices

Total number of active services bound to the vserver.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimeSeconds

Time when last state change happened. Seconds part.

stateChangeTimemSec

Time at which last state change happened. Milliseconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

hits

Number of hits.

piPolicyhits

Number of hits.

AuthenticationHost

Fully qualified domain name (FQDN) of the authentication virtual server to which the user must be redirected for authentication. Make sure that the Authentication parameter is set to ENABLED.

push

Process traffic with the push virtual server that is bound to this load balancing virtual server.

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the load balancing virtual server that you are configuring.

pushLabel

Expression for extracting a label from the server's response. Can be either an expression or the name of a named expression.

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

tcpProfileName

Name of the TCP profile whose settings are to be applied to the virtual server.

httpProfileName

Name of the HTTP profile whose settings are to be applied to the virtual server.

dbProfileName

Name of the DB profile whose settings are to be applied to the virtual server.

comment

Any comments that you might want to associate with the virtual server.

flag

flags

policySubType

l2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to co-exist on the NetScaler appliance.

oracleServerVersion

Oracle server version

mssqlServerVersion

For a load balancing virtual server of type MSSQL, the Microsoft SQL Server version. Set this parameter if you expect some clients to run a version different from the version of the database. This setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

mysqlProtocolVersion

MySQL protocol version that the virtual server advertises to clients.

mysqlServerVersion

MySQL server version string that the virtual server advertises to clients.

mysqlCharacterSet

Character set that the virtual server advertises to clients.

mysqlServerCapabilities

Server capabilities that the virtual server advertises to clients.

appflowLog

Apply AppFlow logging to the virtual server.

netProfile

Name of the network profile to associate with the virtual server. If you set this parameter, the virtual server uses only the IP addresses in the network profile as source IP addresses when initiating connections with servers.

isGslb

This field is set to true if it is a GSLBVserver.

icmpVsrResponse

How the NetScaler appliance responds to ping requests received for an IP address that is common to one or more virtual servers. Available settings function as follows:

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always responds to the ping requests.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance responds to the ping requests if at least one of the virtual servers is UP. Otherwise, the appliance does not respond.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance responds if at least one virtual server with the ACTIVE setting is UP. Otherwise, the appliance does not respond.

Note: This parameter is available at the virtual server level. A similar parameter, ICMP Response, is available at the IP address level, for IPv4 addresses of type VIP. To set that parameter, use the add ip command in the CLI or the Create IP dialog box in the GUI.

newServiceRequest

Number of requests, or percentage of the load on existing services, by which to increase the load on a new service at each interval in slow-start mode. A value of 0 disables slow start for new services, and any services still ramping up. They immediately receive their full share of the load. If you use the CLI, be sure to set the newServiceRequestUnit parameter to specify the unit as number of requests or percentage of load.

newServiceRequestUnit

Units in which to increment load at each interval in slow-start mode.

newServiceRequestIncrementInterval

Interval, in seconds, between successive increments in the load on a new service or a service whose state has just changed from DOWN to UP. A value of 0 (zero) specifies manual slow start.

vsvrcfgflags

Contains the config info of vserver to be used at validation

vsvrbindsvcip

used for showing the ip of bound entities

vsvrbindsvcport

used for showing ports of bound entities

persistAVPno

Persist AVP number for Diameter Persistency.

In case this AVP is not defined in Base RFC 3588 and it is nested inside a Grouped AVP,

define a sequence of AVP numbers (max 3) in order of parent to child. So say persist AVP number X

is nested inside AVP Y which is nested in Z, then define the list as Z Y X

skippersistency

This argument decides the behavior incase the service which is selected from an existing persistence session has reached threshold.

td

Traffic Domain ID

minAutoscaleMembers

Minimum number of members expected to be present when vserver is used in Autoscale.

maxAutoscaleMembers

Maximum number of members expected to be present when vserver is used in Autoscale.

authnProfile

Name of the authentication profile to be used when authentication is turned on.

macmodeRetainvlan

This option is used to retain vlan information of incoming packet when macmode is enabled

dbsLb

For enabling database specific load-balacing..

dns64

This argument is for enabling/disabling the dns64 on lbvserver

bypassAAAA

If this option is enabled while resolving DNS64 query AAAA queries are not sent to back end dns server

RecursionAvailable

When set to YES, this option causes the DNS replies from this vserver to have the RA bit turned on. Typically one would set this option to YES, when the vserver is load balancing a set of DNS servers thatsupport recursive queries.

devno

count

---

stat lb vserver

Displays the statistical data collected for a load balancing virtual server.

Synopsys

stat lb vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)] [-sortBy Hits [<sortOrder>]]

Arguments

name

Name of the virtual server. If no name is provided, statistical data of all configured virtual servers is displayed.

clearstats

Clear the statsistics / counters

Possible values: basic, full

sortBy

use this argument to sort by specific key

Possible values: Hits

Outputs

count

devno

stateflag

Outputs

s surgeQ (vSurgeQ)

Number of requests waiting on this vserver.

Current Client Est connections (ClntEstConn)

Number of client connections in ESTABLISHED state.

total INACTIVE services (inactSvcs)

number of INACTIVE services bound to a vserver

Vserver Health (Health)

Health of the vserver. This gives percentage of UP services bound to this vserver.

Vserver IP address (vsvrIP)

IP address of the vserver

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

total ACTIVE services (actSvcs)

number of ACTIVE services bound to a vserver

Vserver hits (Hits)

Total vserver hits

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Total Packets rcvd (PktRx)

Total number of packets received by this service or virtual server.

Total Packets sent (PktTx)

Total number of packets sent.

Current client connections (ClntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Requests in surge queue (SurgeQ)

Number of requests in the surge queue.

s surgeQs (SvcSurgeQ)

Total number of requests in the surge queues of all the services bound to this LB-vserver.

Spill Over Threshold (SOThresh)

Spill Over Threshold set on the VServer.

Spill Over Hits (NumSo)

Number of times vserver experienced spill over.

Labeled Connection (LblConn)

Number of Labeled connection on this vserver

Push Labeled Connection (PushLbl)

Number of labels for this push vserver.

Deferred Request (DefReq)

Number of deferred request on this vserver

Invalid Request/Response (IvldReqRsp)

Number invalid requests/responses on this vserver

Invalid Request/Response Dropped (IvldReqRspDrp)

Number invalid requests/responses dropped on this vserver

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

---

rename lb vserver

Renames a load balancing virtual server.

Synopsys

rename lb vserver <name>@ <newName>@

Arguments

name

Existing name of the virtual server.

newName

New name for the virtual server.

Example

```
rename lb vserver http_vsvr http_vsvr_new
```

lb wlm

Sep 22, 2015

The following operations can be performed on "lb wlm":

add | rm | set | unset | show | bind | unbind

add lb wlm

Add a Work Load Manager. NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager.

IPAddress

The IP address of the WLM.

LBUID

The LBUID for the Load Balancer to communicate to the Work Load Manager.

KATimeout

The idle time period after which NS would probe the WLM. The value ranges from 1 to 1440 minutes.

Default value: 2

Maximum value: 1440

Example

```
add lb wlm ibm_wlm 10.102.1.10 3060
```

---

rm lb wlm

Removes a Work Load Manager. NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager to be removed.

Example

```
rm lb wlm ibm_wlm
```

---

set lb wlm

set Work Load Manager attributes NOTE: This command is deprecated.

Synopsys

Arguments

wlmName

The name of the work load manager.

KATimeout

The idle time period after which NS would probe the WLM. The value ranges from 1 to 1440 minutes.

Default value: 2

Maximum value: 1440

Example

```
set lb wlm ibm_wlm -ka_timeout 6
```

---

unset lb wlm

Use this command to remove lb wlm settings.Refer to the set lb wlm command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

---

show lb wlm

show Work Load Manager details NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the work load manager.

summary

fullValues

format

level

Outputs

IPAddress

The IP address of the WLM.

port

A port number for the virtual server.

stateflag

secure

Use this parameter to enable secure mode of communication with WLM.

KATimeout

The idle time period after which NS would probe the WLM. The value ranges from 1 to 1440 minutes.

LBUID

The LBUID for the Load Balancer to communicate to the Work Load Manager.

state

State of the WLM.

vServerName

Name of the virtual server which is to be bound to the WLM.

devno

count

Example

show lb wlm ibm_wlm

---

bind lb wlm

Bind a vserver to Work Load Manager. NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager.

vServerName

Name of the virtual server which is to be bound to the WLM.

Example

bind lb wlm ibm_wlm http_vip To bind multiple vservers to workload manager use the following command: bind lb wlm ibm_wlm http_

---

unbind lb wlm

Unbind a vserver from Work Load Manager. NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager.

vServerName

Name of the virtual server which is to be unbound from the WLM.

Example

unbind lb wlm ibm_wlm http_vip To unbind multiple vservers from Work Load Manager use the following command: unbind lb wlm ibm

Networking Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- L2Param
- L3Param
- L4Param
- arp
- arpparam
- bridge
- bridgegroup
- bridgetable
- channel
- ci
- fis
- forwardingSession
- inat
- inatparam
- inatsession
- interface
- ip6Tunnel
- ip6TunnelParam
- ipTunnel
- ipTunnelParam
- ipset
- ipv6
- lacp
- linkset
- nat64
- nd6
- nd6RAvariables
- netProfile
- netbridge
- onLinkIPv6Prefix
- ptp
- rnat
- rnat6
- rnatip
- rnatparam
- route
- route6
- rsskeytype
- tunnelip
- tunnelip6
- vPathParam

- vlan
- vpath
- vrID
- vrID6
- vrIDParam

L2Param

Sep 22, 2015

The following operations can be performed on "L2Param":

set | unset | show

set L2Param

Set Layer 2 related global settings on the NetScaler

Synopsys

set L2Param [-mbfPeermacUpdate <positive_integer>] [-maxBridgeCollision <positive_integer>] [-bdggrpProxyArp (ENABLED | DISABLED)] [-bdgSetting (ENABLED | DISABLED)] [-garpOnVridIntf (ENABLED | DISABLED)] [-macModeFwdMyPkt (ENABLED | DISABLED)] [-useMyMAC (ENABLED | DISABLED)] [-proxyArp (ENABLED | DISABLED)] [-garpReply (ENABLED | DISABLED)] [-mbfInstLearning (ENABLED | DISABLED)] [-rstIntfOnHaFo (ENABLED | DISABLED)] [-skipProxyingBsdTraffic (ENABLED | DISABLED)] [-returnToEthernetSender (ENABLED | DISABLED)]

Arguments

mbfPeermacUpdate

When mbf_instant_learning is enabled, learn any changes in peer's MAC after this time interval, which is in 10ms ticks.

Default value: 10

maxBridgeCollision

Maximum bridge collision for loop detection

Default value: 20

bdggrpProxyArp

Set/reset proxy ARP in bridge group deployment

Possible values: ENABLED, DISABLED

Default value: ENABLED

bdgSetting

Bridging settings for C2C behavior

Possible values: ENABLED, DISABLED

Default value: DISABLED

garpOnVridIntf

Send GARP messagess on VRID-configured interfaces upon failover

Possible values: ENABLED, DISABLED

Default value: ENABLED

macModeFwdMyPkt

MAC mode vserver forward packets destined to VIPs.

Possible values: ENABLED, DISABLED

Default value: DISABLED

useMyMAC

Set/reset cfg_use_my_mac

Possible values: ENABLED, DISABLED

Default value: DISABLED

proxyArp

Set/reset cfg_proxy_arp_dr

Possible values: ENABLED, DISABLED

Default value: ENABLED

garpReply

Set/reset REPLY form of GARP

Possible values: ENABLED, DISABLED

Default value: DISABLED

mbfInstLearning

Enable instant learning of MAC changes in MBF mode.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rstIntfOnHaFo

Enable the reset interface upon HA failover.

Possible values: ENABLED, DISABLED

Default value: DISABLED

skipProxyingBsdTraffic

Enable the proxying of FreeBSD traffic.

Possible values: ENABLED, DISABLED

Default value: DISABLED

returnToEthernetSender

Return to ethernet sender.

Possible values: ENABLED, DISABLED

Default value: DISABLED

---

unset L2Param

Use this command to remove L2Param settings.Refer to the set L2Param command for meanings of the arguments.

Synopsys

unset L2Param [-mbfPeermacUpdate] [-maxBridgeCollision] [-bdggrpProxyArp] [-bdgSetting] [-garpOnVridIntf] [-macModeFwdMyPkt] [-useMyMAC] [-proxyArp] [-garpReply] [-mbfInstLearning] [-rstIntfOnHaFo] [-skipProxyingBsdTraffic] [-returnToEthernetSender]

---

show L2Param

Displays the settings of global Layer 2 parameters on the NetScaler appliance.

Synopsys

show L2Param

Arguments

format

level

Outputs

maxBridgeCollision

Maximum bridge collision for loop detection

linkMTU

this MTU is used for Ready logo purpose, changing the Interface MTU at soft layer level.

mbfPeermacUpdate

When mbf_instant_learning is enabled, learn any changes in peer's MAC after this time interval, which is in

10ms ticks.

bdggrpProxyArp

Set/reset proxy ARP in bridge group deployment

bdgSetting

Bridging settings for C2C behavior

garpOnVridIntf

Send GARP messagess on VRID-configured interfaces upon failover

macModeFwdMyPkt

MAC mode vserver forward packets destined to VIPs.

useMyMAC

Set/reset cfg_use_my_mac

proxyArp

Set/reset cfg_proxy_arp_dr

garpReply

Set/reset REPLY form of GARP

mbfInstLearning

Enable instant learning of MAC changes in MBF mode.

rstIntfOnHaFo

Enable the reset interface upon HA failover.

skipProxyingBsdTraffic

Enable the proxying of FreeBSD traffic.

returnToEthernetSender

Return to ethernet sender.

L3Param

Sep 22, 2015

The following operations can be performed on "L3Param":

set | unset | show

set L3Param

Set Layer 3 related global settings on the NetScaler

Synopsys

set L3Param [-srcnat (ENABLED | DISABLED)] [-icmpGenRateThreshold <positive_integer>] [-overrideRnat (ENABLED | DISABLED)] [-dropDFFlag (ENABLED | DISABLED)] [-mipRoundRobin (ENABLED | DISABLED)] [-externalLoopBack (ENABLED | DISABLED)] [-tnlPmtuWoConn (ENABLED | DISABLED)] [-usipServerStrayPkt (ENABLED | DISABLED)] [-forwardICMPFragments (ENABLED | DISABLED)] [-dropIPFragments (ENABLED | DISABLED)] [-AclLogTime <positive_integer>] [-icmpErrGenerate (ENABLED | DISABLED)] [-implicitACLAllow (ENABLED | DISABLED)]

Arguments

srcnat

Perform NAT if only the source is in the private network

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmpGenRateThreshold

NS generated ICMP pkts per 10ms rate threshold

Default value: 100

overrideRnat

USNIP/USIP settings override RNAT settings for configured

service/virtual server traffic..

Possible values: ENABLED, DISABLED

Default value: DISABLED

dropDFFlag

Enable dropping the IP DF flag.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mipRoundRobin

Enable round robin usage of mapped IPs.

Possible values: ENABLED, DISABLED

Default value: ENABLED

externalLoopBack

Enable external loopback.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tnlPmtuWoConn

Enable external loopback.

Possible values: ENABLED, DISABLED

Default value: ENABLED

usipServerStrayPkt

Enable detection of stray server side pkts in USIP mode.

Possible values: ENABLED, DISABLED

Default value: DISABLED

forwardICMPFragments

Enable forwarding of ICMP fragments.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dropIPFragments

Enable dropping of IP fragments.

Possible values: ENABLED, DISABLED

Default value: DISABLED

AclLogTime

Parameter to tune acl logging time

Default value: 5000

icmpErrGenerate

Enable/Disable fragmentation required icmp error generation, before encapsulating a packet with vPath header. This knob is only functional for vPath Environment

Possible values: ENABLED, DISABLED

Default value: ENABLED

implicitACLAllow

Do not apply ACLs for internal ports

Possible values: ENABLED, DISABLED

Default value: ENABLED

---

unset L3Param

Use this command to remove L3Param settings.Refer to the set L3Param command for meanings of the arguments.

Synopsys

unset L3Param [-srcnat] [-icmpGenRateThreshold] [-overrideRnat] [-dropDFFlag] [-mipRoundRobin] [-externalLoopBack] [-tnlPmtuWoConn] [-usipServerStrayPkt] [-forwardICMPFragments] [-dropIPFragments] [-AclLogTime] [-icmpErrGenerate] [-implicitACLAllow]

---

show L3Param

Displays the settings of global Layer 3 parameters.

Synopsys

show L3Param

Arguments

format

level

Outputs

srcnat

Perform NAT if only the source is in the private network

icmpGenRateThreshold

NS generated ICMP pkts per 10ms rate threshold

overrideRnat

USNIP/USIP settings override RNAT settings for configured

service/virtual server traffic..

dropDFFlag

Enable dropping the IP DF flag.

mipRoundRobin

Enable round robin usage of mapped IPs.

externalLoopBack

Enable external loopback.

tnlPmtuWoConn

Enable external loopback.

usipServerStrayPkt

Enable detection of stray server side pkts in USIP mode.

forwardICMPFragments

Enable forwarding of ICMP fragments.

dropIPFragments

Enable dropping of IP fragments.

AclLogTime

Parameter to tune acl logging time

icmpErrGenerate

Enable/Disable fragmentation required icmp error generation, before encapsulating a packet with vPath header. This knob is only functional for vPath Environment

implicitACLAllow

Do not apply ACLs for internal ports

L4Param

Sep 22, 2015

The following operations can be performed on "L4Param":

set | unset | show

set L4Param

Set Layer 4 related global settings on the NetScaler

Synopsys

set L4Param [-l2ConnMethod <l2ConnMethod>] [-l4switch (ENABLED | DISABLED)]

Arguments

l2ConnMethod

Layer 2 connection method based on the combination of channel number, MAC address and VLAN. It is tuned with l2conn param of lb vserver. If l2conn of lb vserver is ON then method specified here will be used to identify a connection in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>).

Possible values: Channel, Vlan, VlanChannel, Mac, MacChannel, MacVlan, MacVlanChannel

Default value: NS_L2CONN_MAC_VLAN_CHAN

l4switch

In L4 switch topology, always clients and servers are on the same side. Enable l4switch to allow such connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

set l4param

---

unset L4Param

Use this command to remove L4Param settings.Refer to the set L4Param command for meanings of the arguments.

Synopsys

unset L4Param [-l2ConnMethod] [-l4switch]

---

show L4Param

Displays the settings of global Layer 4 parameters.

Synopsys

show L4Param

Arguments

format

level

Outputs

l2ConnMethod

Layer 2 connection method based on the combination of channel number, MAC address and VLAN. It is tuned with l2conn param of lb vserver. If l2conn of lb vserver is ON then method specified here will be used to identify a connection in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>).

l4switch

In L4 switch topology, always clients and servers are on the same side. Enable l4switch to allow such connections.

arp

Sep 22, 2015

The following operations can be performed on "arp":

add | rm | send | show

add arp

Adds a static ARP entry to the ARP table of the NetScaler appliance.

Synopsys

add arp -IPAddress <ip_addr> [-td <positive_integer>] -mac <mac_addr> -ifnum <interface_name> [-ownerNode <positive_integer>]

Arguments

IPAddress

IP address of the network device that you want to add to the ARP table.

td

Traffic Domain Id.

Maximum value: 4094

mac

MAC address of the network device.

ifnum

Interface through which the network device is accessible. Specify the interface in (slot/port) notation. For example, 1/3.

ownerNode

The owner node for the Arp entry.

Default value: VAL_NOT_SET

Maximum value: 31

Example

```
add arp -ip 10.100.0.48 -mac 00:a0:cc:5f:76:3a -ifnum 1/1
```

---

rm arp

Removes a specified static ARP entry or all static ARP entries from the NetScaler appliance's ARP table.

Synopsys

rm arp (<IPAddress> | -all) [-td <positive_integer>] [-ownerNode <positive_integer>]

Arguments

IPAddress

IP address of the network device in the ARP entry that you want to remove from the ARP table.

td

Traffic Domain Id.

Maximum value: 4094

all

Remove all ARP entries from the ARP table of the NetScaler appliance.

ownerNode

The owner node for the Arp entry.

Default value: VAL_NOT_SET

Maximum value: 31

---

send arp

Sends Gratuitous Address Resolution Protocol (GARP) messages for the specified NetScaler owned IP addresses.

Synopsys

send arp ((-IPAddress <ip_addr> [-td <positive_integer>]) | -all)

Arguments

IPAddress

NetScaler owned IP address for which the NetScaler appliance sends Gratuitous Address Resolution Protocol (GARP) messages.

all

Send GARP messages for all NetScaler owned IP addresses on which the ARP option is enabled. In a secondary node of an high availability configuration, this option sends GARP messages for the node's NSIP address only.

Example

send arp 10.10.10.10

---

show arp

Display all the entries in the system's ARP table.

Synopsys

show arp [<IPAddress> [-td <positive_integer>] [-ownerNode <positive_integer>]]

Arguments

IPAddress

The IP address corresponding to an ARP entry.

ownerNode

The cluster node which owns the ARP entry.

Default value: VAL_NOT_SET

Maximum value: 31

summary

fullValues

format

level

Outputs

mac

The MAC address corresponding to an ARP entry.

ifnum

The interface on which this MAC address resides.

timeout

The time, in seconds, after which the entry times out.

state

The state of the ARP entry.

flags

The flags for the entry.

type

Indicates whether this ARP entry was added manually or dynamically. When you manually add an ARP entry, the value for this parameter is STATIC. Otherwise, it is DYNAMIC. For the NSIP and loopback IP addresses, the value is PERMANENT.

vlan

The VLAN ID through which packets are to be sent after matching the ARP entry. This is a numeric value.

channel

The tunnel, channel, or physical interface through which the ARP entry is identified.

flag

Flags for the entry.

devno

count

stateflag

Example

The output of the sh arp command is as follows: 5 configured arps: IP MAC Iface TD VLAN Origin TTL -------

arpparam

Sep 22, 2015

The following operations can be performed on "arpparam":

set | unset | show

set arpparam

Sets a global time-out value for dynamic ARP entries.

Synopsys

set arpparam [-timeout <positive_integer>] [-spoofValidation (ENABLED | DISABLED)]

Arguments

timeout

Time-out value (aging time) for the dynamically learned ARP entries, in seconds. The new value applies only to ARP entries that are dynamically learned after the new value is set. Previously existing ARP entries expire after the previously configured aging time.

Default value: 1200

Minimum value: 5

Maximum value: 1200

spoofValidation

enable/disable arp spoofing validation

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set arpparam -timeout 200 -spoofvalidate ENABLE
```

---

unset arpparam

Use this command to remove arpparam settings.Refer to the set arpparam command for meanings of the arguments.

Synopsys

unset arpparam [-timeout] [-spoofValidation]

---

show arpparam

Display the global setting of dynamically learned ARP entries.

Synopsys

show arpparam

Arguments

format

level

Outputs

timeout

The ARP table entry aging time, in seconds.

spoofValidation

enable/disable arp spoofing validation

Example

```
show arpparam
```

bridge

Sep 22, 2015

The following operations can be performed on "bridge":

stat bridge

Display bridging statistics.

Synopsys

stat bridge [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

Loops

The number of times bridging registered MAC moved

Collisions (Collisns)

The number of bridging table collisions

Interface muted (Mutes)

The number of bridging related interface mutes

Total bridged packets (Tot_pkts)

The total number of bridged packets

Total bridged Mbits (Tot_Mbits)

The total number of bridged Mbits

bridgegroup

Sep 22, 2015

The following operations can be performed on "bridgegroup":

add | rm | set | unset | bind | unbind | show

add bridgegroup

Create a Bridge group.

Synopsys

add bridgegroup <id> [-ipv6DynamicRouting (ENABLED | DISABLED)]

Arguments

id

An integer that uniquely identifies the bridge group. Minimum value: 1. Maximum value: 1000.

Minimum value: 1

Maximum value: 1000

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this VLAN. Possible values: ENABLED, DISABLED Default: DISABLED. Note: For the ENABLED setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line. For more information about configuring IPv6 dynamic routing protocols on the NetScaler appliance, see the 'Dynamic Routing' chapter of the Citrix NetScaler Networking Guide.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add bridgegroup bg1
```

---

rm bridgegroup

Remove the bridge group created by the add bridge group command.

Synopsys

rm bridgegroup <id>

Arguments

id

An integer that uniquely identifies the bridge group that you want to remove from the NetScaler appliance.

Minimum value: 1

Maximum value: 1000

---

set bridgegroup

Set Bridge group parameters.

Synopsys

set bridgegroup <id> -ipv6DynamicRouting (ENABLED | DISABLED)

Arguments

id

An integer value that uniquely identifies the bridge group. Minimum value: 1. Maximum value: 1000.

Minimum value: 1

Maximum value: 1000

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this bridge group. For this setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line. For more information about configuring IPv6 dynamic routing protocols on the NetScaler appliance, see the Dynamic Routing chapter of the Citrix NetScaler Networking Guide.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set bridgegroup bg1 -dynamicRouting ENABLED
```

---

unset bridgegroup

Use this command to remove bridgegroup settings.Refer to the set bridgegroup command for meanings of the arguments.

Synopsys

```
unset bridgegroup <id> -ipv6DynamicRouting
```

---

bind bridgegroup

Bind a vlan or an ip address to a bridgegroup.

Synopsys

```
bind bridgegroup <id> [-vlan <positive_integer>] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]
```

Arguments

id

The integer that uniquely identifies the bridge group.

Minimum value: 1

Maximum value: 1000

vlan

An integer that uniquely identifies the VLAN that you want to bind to this bridge group.

Minimum value: 2

Maximum value: 4094

IPAddress

A network address or addresses to be associated with the bridge group. You must add entries for these network addresses in the routing table before running this command.

Example

```
bind bridgegroup bg1 -vlan 2
```

---

unbind bridgegroup

Unbinds the specified VLANs or IP addresses from a bridge group.

Synopsys

```
unbind bridgegroup <id> [-vlan <positive_integer>] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]
```

Arguments

id

Integer that uniquely identifies the bridge group.

Minimum value: 1

Maximum value: 1000

vlan

ID of the VLAN to unbind from this bridge group.

Minimum value: 2

Maximum value: 4094

IPAddress

Network address associated with the bridge group.

---

show bridgegroup

Display the configured bridge group. If a name is specified, only that particular bridge group information is displayed. Otherwise, all configured bridge groups are displayed.

Synopsys

show bridgegroup [<id>]

Arguments

id

The name of the bridge group.

Minimum value: 1

Maximum value: 1000

summary

fullValues

format

level

Outputs

td

Traffic Domain Id.

IPAddress

The IP address assigned to the bridge group.

netmask

The network mask for the subnet defined for the bridge group.

flags

Temporary flag used for internal purpose.

portbitmap

Member interfaces of this bridge group.

tagbitmap

Tagged members of this bridge group.

ifaces

Names of all member interfaces of this bridge group.

tagIfaces

Names of all tagged member interfaces of this bridge group.

vlan

Names of all member VLANs.

ipv6DynamicRouting

Whether dynamic routing is enabled or disabled.

rnat

Temporary flag used for internal purpose.

flag

devno

count

stateflag

Example

An example of the output of the show bridge group command is as follows: 2 configured Bridge Group: 1) Bridge Group: 1

bridgetable

Sep 22, 2015

The following operations can be performed on "bridgetable":

set | unset | show | clear

set bridgetable

Sets global parameters of bridge table entries.

Synopsys

set bridgetable -bridgeAge <positive_integer>

Arguments

bridgeAge

Time-out value for the bridge table entries, in seconds. The new value applies only to the entries that are dynamically learned after the new value is set. Previously existing bridge table entries expire after the previously configured time-out value.

Default value: 300

Minimum value: 60

Maximum value: 300

Example

```
set bridgetable -bridgeAge 200
```

---

unset bridgetable

Use this command to remove bridgetable settings.Refer to the set bridgetable command for meanings of the arguments.

Synopsys

unset bridgetable -bridgeAge

---

show bridgetable

Displays the bridge table entries and the configured time-out values for these entries.

Synopsys

show bridgetable

Arguments

summary

fullValues

format

level

Outputs

bridgeAge

Time-out value for the bridge table entries, in seconds. The new value applies only to the entries that are dynamically learned after the new value is set. Previously existing bridge table entries expire after the previously configured time-out value.

mac

The MAC address of the target.

ifnum

The interface on which the address was learned.

vlan

The VLAN in which this MAC address resides.

channel

The Tunnel through which bridge entry is learned.

devno

count

stateflag

Example

show bridgetable

---

clear bridgetable

Remove entries from bridge table

Synopsys

clear bridgetable [-vlan <positive_integer>] [-ifnum <interface_name>]

Arguments

vlan

VLAN whose entries are to be removed.

ifnum

INTERFACE whose entries are to be removed.

channel

Sep 22, 2015

The following operations can be performed on "channel":

add | rm | set | unset | bind | unbind | show

add channel

Creates a link aggregate channel on the NetScaler appliance or on a cluster configuration. Link aggregation combines data coming from multiple ports into a single high-speed link. Configuring link aggregation increases the capacity and availability of the communication channel between the NetScaler appliance and other connected devices. When a network interface is bound to a channel, the channel parameters have precedence over the network interface parameters. That is, the network interface parameters are ignored. A network interface can be bound only to one channel.

Synopsys

add channel <id> [-ifnum <interface_name> ...] [-state (ENABLED | DISABLED)] [-lamac <mac_addr>] [-speed <speed>] [-flowControl <flowControl>] [-haMonitor (ON | OFF)] [-tagall (ON | OFF)] [-ifAlias <string>] [-throughput <positive_integer>] [-bandwidthHigh <positive_integer> [-bandwidthNormal <positive_integer>]]

Arguments

id

ID for the LA channel or cluster LA channel to be created. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or cluster LA channel in CLA/x notation, where x can range from 1 to 4. Cannot be changed after the LA channel is created.

ifnum

Interfaces to be bound to the LA channel of a NetScaler appliance or to the LA channel of a cluster configuration.

For an LA channel of a NetScaler appliance, specify an interface in C/U notation (for example, 1/3).

For an LA channel of a cluster configuration, specify an interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

state

Enable or disable the LA channel.

Possible values: ENABLED, DISABLED

Default value: NSA_DVC_ENABLE

Mode

The initital mode for the LA channel.

Possible values: MANUAL, AUTO

connDistr

The 'connection' distribution mode for the LA channel.

Possible values: DISABLED, ENABLED

macdistr

The 'MAC' distribution mode for the LA channel.

Possible values: SOURCE, DESTINATION, BOTH

lamac

MAC address for LA channels on VPX Platforms namely VPX on SDX,Xen,ESX.

speed

Ethernet speed of the channel, in Mbps. If the speed of any bound interface is greater than or equal to the value set for this parameter, the state of the interface is UP. Otherwise, the state is INACTIVE. Bound Interfaces whose state is INACTIVE do not process any traffic.

Possible values: AUTO, 10, 100, 1000, 10000, 40000

Default value: NSA_DVC_SPEED_AUTO

flowControl

Flow control for the LA channel.

Possible values: OFF, RX, TX, RXTX

Default value: NSA_DVC_FC_OFF

haMonitor

In a High Availability (HA) configuration, monitor the LA channel for failure events. Failure of any LA channel that has HA MON enabled triggers HA failover.

Possible values: ON, OFF

Default value: NSA_DVC_MONITOR_ON

tagall

Adds a four-byte 802.1q tag to every packet sent on this channel. The ON setting applies tags for all VLANs that are bound to this channel. OFF applies the tag for all VLANs other than the native VLAN.

Possible values: ON, OFF

Default value: NSA_DVC_VTRUNK_OFF

trunk

This is deprecated by tagall

Possible values: ON, OFF

Default value: OFF

ifAlias

Alias name for the LA channel. Used only to enhance readability. To perform any operations, you have to specify the LA channel ID.

Default value: " "

throughput

Low threshold value for the throughput of the LA channel, in Mbps. In an HA configuration, failover is triggered if the LA channel has HA MON enabled and the throughput is below the specified threshold.

Maximum value: 80000

bandwidthHigh

High threshold value for the bandwidth usage of the LA channel, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the LA channel is greater than or equal to the specified high threshold value.

Maximum value: 80000

---

rm channel

Removes an LA channel from the NetScaler appliance or a cluster LA channel from a cluster configuration. Important: When a LA channel is removed, the network interfaces bound to it induce network loops that decrease network performance. You must disable the network interfaces before you remove the channel.

Synopsys

rm channel <id>

Arguments

id

ID of the LA channel or cluster LA channel that you want to remove. Specify an LA channel in LA/x notation, where

x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

---

set channel

Modifies the specified parameters of an LA channel.

Synopsys

set channel <id> [-state (ENABLED | DISABLED)] [-lamac <mac_addr>] [-speed <speed>] [-flowControl <flowControl>] [-haMonitor (ON | OFF)] [-tagall (ON | OFF)] [-ifAlias <string>] [-throughput <positive_integer>] [-lrMinThroughput <positive_integer>] [-bandwidthHigh <positive_integer> [-bandwidthNormal <positive_integer>]]

Arguments

id

ID of the LA channel or the cluster LA channel whose parameters you want to modify. Specify an LA channel in LA/x, where x can range from 1 to 8 notation or a cluster LA channel in CLA/x notation, where x can range from 1 to 4. Required for identifying the LA channel and cannot be modified.

state

Enable or disable the LA channel.

Possible values: ENABLED, DISABLED

Default value: NSA_DVC_ENABLE

Mode

The mode for the LA channel.

Possible values: MANUAL, AUTO

connDistr

The 'connection' distribution mode for the LA channel.

Possible values: DISABLED, ENABLED

macdistr

The 'MAC' distribution mode for the LA channel.

Possible values: SOURCE, DESTINATION, BOTH

lamac

Allows User to set MAC address for LA channels on Hypervised platforms.

speed

The speed for the LA channel.

Possible values: AUTO, 10, 100, 1000, 10000, 40000

Default value: NSA_DVC_SPEED_AUTO

flowControl

Required flow control for the LA channel.

Possible values: OFF, RX, TX, RXTX

Default value: NSA_DVC_FC_OFF

haMonitor

The state of HA monitoring for the LA channel.

Possible values: ON, OFF

Default value: NSA_DVC_MONITOR_ON

tagall

The appliance adds a four-byte 802.1q tag to every packet sent on this channel. ON applies tags for all the VLANs that are bound to this channel. OFF, applies the tag for all VLANs other than the native VLAN.

Possible values: ON, OFF

Default value: NSA_DVC_VTRUNK_OFF

trunk

This is deprecated by tagall.

Possible values: ON, OFF

Default value: OFF

ifAlias

The alias name for the interface.

Default value: " "

throughput

Low threshold value for the throughput of the LA channel, in Mbps. In an HA configuration, failover is triggered if the LA channel has HA MON enabled and the throughput is below the specified threshold.

Maximum value: 80000

lrMinThroughput

Minimum required throughput for a channel where we require Link Redundancy. When throughput falls below the threshold, the subset of interfaces which can give maximum throughput will become active When configured in HA pair, this will work along with the throughput parameter set. If lrMinThroughput can be achieved Link Failover will be

attempted before Node Failover

Maximum value: 80000

bandwidthHigh

High threshold value for the bandwidth usage of the LA channel, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the LA channel is greater than or equal to the specified high threshold value.

Maximum value: 80000

---

unset channel

Use this command to remove channel settings.Refer to the set channel command for meanings of the arguments.

Synopsys

unset channel <id> [-state] [-speed] [-flowControl] [-haMonitor] [-tagall] [-ifAlias] [-throughput] [-lrMinThroughput] [-bandwidthHigh] [-bandwidthNormal]

---

bind channel

Binds the specified interfaces to a channel.

Synopsys

bind channel <id> <ifnum> ...

Arguments

id

ID of the LA channel or the cluster LA channel to which you want to bind interfaces. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

ifnum

Interfaces to be bound to the LA channel of a NetScaler appliance or to the LA channel of a cluster configuration.

For an LA channel of a NetScaler appliance, specify an interface in C/U notation (for example, 1/3).

For an LA channel of a cluster configuration, specify an interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

---

unbind channel

Unbinds the specified interfaces from an LA channel.

Synopsys

unbind channel <id> <ifnum> ...

Arguments

id

ID of the LA channel or cluster LA channel from which you want to unbind interfaces. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

ifnum

Interfaces to be unbound from the LA channel of a NetScaler appliance or from the LA channel of a cluster configuration.

For an LA channel of a NetScaler appliance, specify an interface in C/U notation (for example, 1/3).

For an LA channel of a cluster configuration, specify an interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

---

show channel

Displays the settings of all LA channels or of the specified channel. To display the settings of all channels, run the command without any parameters. To display the settings of a particular channel, specify the ID of the channel.

Synopsys

show channel [<id>]

Arguments

id

ID of an LA channel or LA channel in cluster configuration whose details you want the NetScaler appliance to display.

Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

Minimum value: 1

summary

fullValues

format

level

Outputs

stateflag

deviceName

LA channel name in form LA/x, where x is channel ID, which ranges from 1 to 8.

unit

Unit number of the channel. This is an internal reference number that the NetScaler uses to identify the channel.

description

The IEEE standard that the channel is based on.

flags

Flags of this channel.

mtu

MTU of the channel. This is the maximum frame size that the channel can process.

vlan

Native VLAN of the channel.

mac

MAC address of the channel.

lamac

MAC address for LA channels on VPX Platforms namely VPX on SDX,Xen,ESX.

uptime

Duration for which the channel is UP. (Example: 3 hours 1 minute 1 second). This value is reset when the channel state changes to DOWN.

downTime

Duration for which the channel is DOWN. (Example: 3 hours 1 minute 1 second). This value is reset when the channel state changes to UP.

reqMedia

Requested media setting for this channel. Since there is no media associated with LA, the displayed values carry no significance.

reqSpeed

Requested speed setting for this channel. Since no media are associated with LA, this speed is used to determine the threshold for the slave interfaces. If the speed of the member interface is less than the requested speed, that interface is considered inactive.

reqDuplex

Requested duplex setting for this channel. Since no media are associated with LA, the displayed values carry no significance.

reqFlowcontrol

Requested flow control setting for this channel. Since no media are associated with LA, the displayed values carry no significance.

media

Requested media setting for this interface.

speed

Actual speed setting for this channel.

duplex

Actualduplex setting for this interface.

flowControl

Actual flow control setting for this channel.

connDistr

Connection distribution setting on this Channel.

macdistr

MAC distribution setting on this Channel.

Mode

The mode(AUTO/MANNUAL) for the LA channel.

haMonitor

HA monitoring enabled or disabled for this channel.

state

Enable or disable the LA channel.

autoneg

Requested auto negotiation setting for this channel. Since no media are associated with LA, this setting has no effect.

autonegResult

Actual auto negotiation setting for this channel.

tagged

VLAN tags setting on this channel.

tagall

The appliance adds a four-byte 802.1q tag to every packet sent on this channel. ON applies tags for all the VLANs that are bound to this channel. OFF, applies the tag for all VLANs other than the native VLAN.

trunk

This is deprecated by tagallNOTE: This attribute is deprecated.The "trunk" argument is confused with LA-trunk, renaming this to "tagall" instead.

taggedAny

Channel setting to accept/drop all tagged packets.

taggedAutolearn

Dynaminc vlan membership on this channel.

hangDetect

Hang detect for this channel.

hangReset

Hang reset for this channel.

linkState

The current state of the link associated with the interface. For logical interfaces (LA), the state of the link is dependent on the state of the slave interfaces. For the link to be UP at least one of the slave interfaces needs to be UP.

intfState

Current state of the specified interface. The interface state set to UP only if the link state is UP and administrative state is ENABLED.

rxpackets

Number of bytes received by all the slave interfaces of the channel since the NetScaler appliance was started or the interface statistics were cleared.

rxbytes

Number of packets received by all member interfaces since the NetScaler appliance was started or the interface statistics were cleared.

rxerrors

Number of inbound packets dropped by the hardware of the slave interfaces since the NetScaler appliance was started or the interface statistics were cleared. Possible causes of dropped packets are CRC, length (undersize or oversize), and alignment errors.

rxdrops

Number of inbound packets dropped by the channel's slave interfaces. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. In most healthy networks, this statistic increments at a steady rate regardless of traffic load. A sharp spike in dropped packets generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

txpackets

Number of packets transmitted by slave interfaces of a channel since the NetScaler appliance was started or the interface statistics were cleared.

txbytes

Number of bytes transmitted by slave interfaces of a channel since the NetScaler appliance was started or the interface statistics were cleared.

txerrors

Number of outbound packets dropped by the hardware of a channel's slave interfaces since the NetScaler appliance was started or the interface statistics

were cleared. Possible causes of dropped packets are length (undersize or oversize) errors and lack of resources.

txdrops

Number of packets dropped in transmission by a channel's slave interfaces for one of the following reasons:

(1) VLAN mismatch.

(2) Oversized packets.

(3) Interface congestion.

(4) Loopback packets sent on non-loopback interface.

inDisc

Number of error-free inbound packets discarded by a channel's slave interfaces because of a lack of resources (for example, insufficient receive buffers).

outDisc

Number of error-free outbound packets discarded by a channel's slave interfaces because of a lack of resources. This statistic is not available on:

(1) 10G ports of NetScaler MPX 12500/12500/15500-10G platforms.

(2) 10G data ports on NetScaler MPX 17500/19500/21500 platforms.

fctls

Number of times flow control is performed on a channel's slave interfaces because of pause frames.

hangs

Number of hangs that occurred on the channel's slave interfaces.

stsStalls

Number of status stalls that occurred on the channel's slave interfaces.

txStalls

Number of Tx stalls happened that occurred on the channel's slave interfaces.

rxStalls

Number of Rx stalls that occurred on the channel's slave interfaces.

bdgMuted

Number of times a channel's slave interfaces stopped transmitting and receiving packets because of MAC moves between ports.

vmac

Virtual MAC of this channel.

vmac6

Virtual MAC for IPv6 on this interface.

ifAlias

The alias name for the interface.

reqThroughput

Minimum required throughput for an interface. Failover is triggered if the operating throughput of a Link Aggregation (LA) channel for which HAMON is ON falls below this value.

lrMinThroughput

Minimum required throughput for a channel where we require Link Redundancy. When throughput falls below the threshold, the subset of interfaces which can give maximum throughput will become active When configured in HA pair, this will work along with the throughput parameter set. If lrMinThroughput can be achieved Link Failover will be attempted before Node Failover

throughput

Actual throughput for the interface.

bandwidthHigh

High threshold value for the bandwidth usage of the LA channel, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the LA channel is greater than or equal to the specified high threshold value.

bandwidthNormal

Normal threshold value for the bandwidth usage of the LA channel, in Mbps. When the bandwidth usage of the LA channel becomes less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

ifnum

The interfaces bound to link aggregate channel.

backplane

The cluster backplane status of the LA. If the status is enabled, the LA is part of the cluster backplane. By default, the backplane status is disabled.

clearTime

Time since the interface stats are cleared last time.

slavestate

State of the member interfaces.

slavemedia

Media type of the member interfaces.

slavespeed

Speed of the member interfaces.

slaveduplex

Duplex of the member interfaces.

slaveflowctl

Flowcontrol of the member interfaces.

slavetime

UP time of the member interfaces.

lacpMode

The LACP mode of the specified interface. The possible values are:

1. Active: A port in active mode generates LACP protocol messages on a regular basis, regardless of any need expressed by its partner to receive them.

2. Passive: A port in passive mode generally does not transmit LACP messages unless its partner is in the active mode; that is, it does not speak unless spoken to.

3. Disabled: Removes the interface from the LA channel. If this is only interface in the LA channel, the LA channel is also deleted.

lacpTimeout

Time to wait for the LACPDU. If a LACPDU is not received within this interval, the NetScaler markes the link partner port as DOWN. Possible values: Long and Short. Long lacptimeout is 90 sec and Short LACP timeout is 3 sec.

lacpActorPriority

LACP Actor Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier. The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lacpActorPortno

LACP Actor port number. LACP uses the port priority with the port number to form the port identifier.

lacpPartnerState

LACP Partner State. Whether the port is in Active or Passive negotiating state.

lacpPartnerTimeout

The timeout value for the information revieved in LACPDUs. It can have values as SHORT or LONG. The SHORT timeout is 3s and the LONG timeout is 90s.

lacpPartnerAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lacpPartnerInsync

The Synchronization flag indicates that the transmitting participant.s mux component is in sync with the system id and key information transmitted.

lacpPartnerCollecting

The Collecting flag indicates that the participant.s collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lacpPartnerDistributing

The Distributing flag indicates that the participant.s distributor is not definitely off. If reset the flag indicates not distributing.

lacpPartnerDefaulted

If the timer expires in the Expired state, the Receive Machine enters the Defaulted state.

lacpPartnerExpired

If the LACPDUs are received for timeout period, the Receive Machine enters the Expired state and the timer is restarted with the timeout value of SHORT timeout

lacpPartnerPriority

LACP Partner Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier.

The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lacpPartnerSystemMac

LACP Partner System MAC.

lacpPartnerSystemPriority

LACP Partner System Priority. The LACP partner's system priority. The values for the priority range from 0 to 65535. The lower the value, the higher the system priority. The switch with the lower system priority value determines which links between LACP partner are active and which are in the standby for each LACP Channel.

lacpPartnerPortno

LACP Partner Port number. LACP uses the port priority with the port number to form the port identifier.

lacpPartnerKey

LACP Partner Key. The LACP key used by the partner port.

lacpActorAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lacpActorInsync

The Synchronization flag indicates that the transmitting participant.s mux component is in sync with the system id and key information transmitted.

lacpActorCollecting

The Collecting flag indicates that the participant.s collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lacpActorDistributing

The Distributing flag indicates that the participant.s distributor is not definitely off. If reset the flag indicates not distributing.

lacpPortMuxState

LACP Port MUX state. The state of the MUX control machine. The Mux Control Machine attaches the physical port to an aggregate port, using the Selection Logic to choose an appropriate port, and turns the distributor and collector for the physical port on or off as required by protocol information.

lacpPortRxStat

LACP Port RX state. The state of the Receive machine. The Receive Machine maintains partner information, recording protocol information from LACPDUs

sent by remote partner(s). Received information is subject to a timeout, and if sufficient time elapses the receive machine will revert to using default partner information.

lacpPortSelectState

LACP Port SELECT state. The state of the SELECT state machine, It could be SELECTED or UNSELECTED.

devno

count

ci

Sep 22, 2015

The following operations can be performed on "ci":

show ci

Displays all the critical interfaces of the NetScaler appliance. In a High Availability configuration, an interface that has HA MON enabled and is not bound to any FIS, is a critical interface. Failure of any critical interface triggers HA failover.

Synopsys

show ci

Arguments

summary

fullValues

Outputs

ifaces

Interfaces that are critical for the appliance to operate in high availability mode.

devno

count

stateflag

Example

```
>show ci Critical Interfaces: LO/1 1/2
```

fis

Sep 22, 2015

The following operations can be performed on "fis":

add | rm | bind | unbind | show

add fis

Adds a failover interface set (FIS) to the NetScaler appliance. A FIS is a logical group of interfaces. In an HA configuration, using a FIS is a way to prevent failover by grouping interfaces so that, when one interface fails, other functioning interfaces are still available.

Synopsys

add fis <name>

Arguments

name

Name for the FIS to be created. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ( ).

---

rm fis

Removes an FIS from the NetScaler appliance. When an FIS is removed, its interfaces are marked as critical interfaces.

Synopsys

rm fis <name>

Arguments

name

Name of the FIS that you want to remove from the NetScaler appliance.

---

bind fis

Binds the specified interfaces to a FIS.

Synopsys

bind fis <name> <ifnum> ...

Arguments

name

The name of the FIS to which you want to bind interfaces.

ifnum

Interface to be bound to the FIS, specified in slot/port notation (for example, 1/3).

---

unbind fis

Unbinds the specified interfaces from a FIS. An unbound interface becomes a critical interface if it is enabled and HA MON is on.

Synopsys

unbind fis <name> <ifnum> ...

Arguments

name

Name of the FIS from which to unbind interfaces.

ifnum

Interfaces to unbind from the FIS, specified in slot/port notation (for example, 1/3). Use spaces to separate multiple entries.

---

show fis

Displays the configured FISs.

Synopsys

show fis [<name>]

Arguments

name

The name of the FIS configured on the appliance.

summary

fullValues

format

level

Outputs

ifaces

Interfaces to be bound to the FIS, in slot/port notation (for example, 1/3).

devno

count

stateflag

Example

>show fis 1) FIS: fis1 Member Interfaces : 1/1 Done

forwardingSession

Sep 22, 2015

The following operations can be performed on "forwardingSession":

add | set | rm | show

add forwardingSession

Adds a forwarding session rule, which creates forwarding-session entries for traffic that originates from or is destined for a particular network and is forwarded by the NetScaler appliance. By default, the appliance does not create session entries for traffic that only forwards (L3 mode). Add a forwarding session rule for a case in which a client request that the appliance forwards to a server results in a response that has to return by the same path

Synopsys

add forwardingSession <name> ((<network> [<netmask>]) | -acl6name <string> | -aclname <string>) [-connfailover (ENABLED | DISABLED)]

Arguments

name

Name for the forwarding session rule. Can begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rule" or 'my rule').

network

Network address from which the forwarded traffic originates or to which it is destined.

acl6name

Name of any configured ACL6 whose action is ALLOW. The rule of the ACL6 is used as an forwarding session rule.

aclname

Name of an extended ACL with action set to ALLOW. The rule specified in the ACL is used as a forwarding-session rule.

connfailover

Synchronize connection information with the secondary appliance in a high availability (HA) pair. That is, synchronize all connection-related information for the forwarding session.

Possible values: ENABLED, DISABLED

Default value: DISABLED

set forwardingSession

Modifies parameters of a forwarding session rule.

Synopsys

set forwardingSession <name> [-connfailover (ENABLED | DISABLED)]

Arguments

name

Name of the forwarding session rule. Required for identifying the forwarding session rule.

connfailover

Synchronize connection information with the secondary appliance in a high availability (HA) pair. That is, synchronize all connection-related information for the forwarding session.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

set forwardsession fw1 -connfailover enabled.

rm forwardingSession

Removes a forwarding session rule from the NetScaler appliance.

Synopsys

rm forwardingSession <name>

Arguments

name

Name of the forwarding session rule to be removed.

Example

rm forwardsession name.

show forwardingSession

Displays the settings of all forwarding session rules configured on the NetScaler appliance, or of the specified forwarding session rule.

Synopsys

show forwardingSession [<name>]

Arguments

name

Name of the forwarding session rule whose details you want to display.

format

level

Outputs

network

Network address from which the forwarded traffic originates or to which it is destined.

netmask

Subnet mask associated with the network.

aclname

Name of an extended ACL with action set to ALLOW. The rule specified in the ACL is used as a forwarding-session rule.

acl6name

Name of any configured ACL6 whose action is ALLOW. The rule of the ACL6 is used as an forwarding session rule.

connfailover

Synchronize connection information with the secondary appliance in a high availability (HA) pair. That is, synchronize all connection-related information for the forwarding session.

devno

count

stateflag

inat

Sep 22, 2015

The following operations can be performed on "inat":

add | rm | set | unset | stat | show

add inat

Adds an INAT rule to the NetScaler appliance. When a packet generated by a client matches the conditions specified in the INAT rule, the appliance translates the packet's public destination IP address to a private destination IP address and forwards the packet to the server at that address.

Synopsys

add inat <name>@ <publicIP>@ <privateIP>@ [-tcpproxy (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-tftp (ENABLED | DISABLED)] [-usip (ON | OFF)] [-usnip (ON | OFF)] [-proxyIP <ip_addr|ipv6_addr>] [-mode STATELESS] [-td <positive_integer>]

Arguments

name

Name for the Inbound NAT (INAT) entry. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ( ).

publicIP

Public IP address of packets received on the NetScaler appliance. Can be aNetScaler-owned VIP or VIP6 address.

privateIP

IP address of the server to which the packet is sent by the NetScaler. Can be an IPv4 or IPv6 address.

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ftp

Enable the FTP protocol on the server for transferring files between the client and the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tftp

To enable/disable TFTP (Default DISABLED).

Possible values: ENABLED, DISABLED

Default value: DISABLED

usip

Enable the NetScaler appliance to retain the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: OFF

usnip

Enable the NetScaler appliance to use a SNIP address as the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: ON

proxyIP

Unique IP address used as the source IP address in packets sent to the server. Must be a MIP or SNIP address.

mode

Stateless translation.

Possible values: STATELESS

td

Traffic Domain Id.

Maximum value: 4094

Example

```
add nat mynat 1.2.3.4 192.168.1.100
```

---

rm inat

Remove the specified Inbound NAT configuration.

Synopsys

```
rm inat <name>@
```

Arguments

name

Name of the Inbound NAT entry to be removed from the NetScaler appliance.

Example

rm nat mynat.

---

set inat

Modifies parameters of an INAT rule.

Synopsys

set inat <name>@ [-privateIP <ip_addr|ipv6_addr>@] [-tcpproxy (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-tftp (ENABLED | DISABLED)] [-usip (ON | OFF)] [-usnip (ON | OFF)] [-proxyIP <ip_addr|ipv6_addr>] [-mode STATELESS]

Arguments

name

The name of the Inbound NAT (INAT) entry that you want to modify.

privateIP

IP address of the server to which the packet is sent by the NetScaler. Can be an IPv4 or IPv6 address.

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ftp

Enable the FTP protocol on the server for transferring files between the client and the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tftp

To enable/disable TFTP (Default DISABLED).

Possible values: ENABLED, DISABLED

Default value: DISABLED

usip

Enable the NetScaler appliance to retain the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: OFF

usnip

Enable the NetScaler appliance to use a SNIP address as the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: ON

proxyIP

A unique IP address used as the source IP address in packets sent to the server. Must be a MIP or SNIP address.

mode

Stateless translation.

Possible values: STATELESS

Example

```
set nat mynat -tcpproxy ENABLED
```

---

unset inat

Use this command to remove inat settings.Refer to the set inat command for meanings of the arguments.

Synopsys

```
unset inat <name>@ [-tcpproxy] [-ftp] [-tftp] [-usip] [-usnip] [-proxyIP] [-mode]
```

---

stat inat

Display statistics for inat sessions.

Synopsys

```
stat inat [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

The INAT.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

TCP Packets translated (V4->V6) (nat46TotTcp46)

Total TCP packets translated (V4->v6).

UDP Packets translated (V4->V6) (nat46TotUdp46)

Total UDP packets translated (V4->v6).

ICMP Packets translated (V4->V6) (nat46TotIcmp46)

Total ICMP packets translated (V4->v6).

Total IPV4 packets dropped (nat46Totdrop46)

Total IPV4 packets dropped.

TCP Packets translated (V6->V4) (nat46TotTcp64)

Total TCP packets translated (V6->v4).

UDP Packets translated (V6->V4) (nat46TotUdp64)

Total UDP packets translated (V6->v4).

ICMP Packets translated (V6->V4) (nat46TotIcmp64)

Total ICMP packets translated (V6->v4).

Total IPV6 packets dropped (nat46Totdrop64)

Total IPV6 packets dropped.

TCP Packets translated (V4->V6) (inatNat46Tcp46)

TCP packets translated (V4->v6).

UDP Packets translated (V4->V6) (inatNat46Udp46)

UDP packets translated (V4->v6).

ICMP Packets translated (V4->V6) (inatNat46Icmp46)

ICMP packets translated (V4->v6).

IPV4 packets dropped (inatNat46drop46)

IPV4 packets dropped.

TCP Packets translated (V6->V4) (inatNat46Tcp64)

TCP packets translated (V6->v4).

UDP Packets translated (V6->V4) (inatNat46Udp64)

UDP packets translated (V6->v4).

ICMP Packets translated (V6->V4) (inatNat46Icmp64)

ICMP packets translated (V6->v4).

IPV6 packets dropped (inatNat46drop64)

IPV6 packets dropped.

Example

stat inat

---

show inat

show all configured inbound NAT.

Synopsys

show inat [<name>]

Arguments

name

Name for the Inbound NAT (INAT) entry. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ( ).

summary

fullValues

format

level

Outputs

publicIP

Public IP address of packets received on the NetScaler appliance. Can be aNetScaler-owned VIP or VIP6 address.

privateIP

IP address of the server to which the packet is sent by the NetScaler. Can be an IPv4 or IPv6 address.

proxyIP

Source IP address for connection to a server.

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

ftp

Enable the FTP protocol on the server for transferring files between the client and the server.

tftp

To enable/disable TFTP (Default DISABLED).

usip

Enable the NetScaler appliance to retain the source IP address of packets before sending the packets to the server.

usnip

Enable the NetScaler appliance to use a SNIP address as the source IP address of packets before sending the packets to the server.

flags

Flags for different modes

mode

Stateless translation.

td

Traffic Domain Id.

devno

count

stateflag

Example

show nat

inatparam

Sep 22, 2015

The following operations can be performed on "inatparam":

set | unset | show

set inatparam

Set the inat parameter

Synopsys

set inatparam [-nat46v6Prefix <ipv6_addr|*>] [-nat46IgnoreTOS (YES | NO)] [-nat46ZeroCheckSum (ENABLED | DISABLED)] [-nat46v6Mtu <positive_integer>] [-nat46FragHeader (ENABLED | DISABLED)]

Arguments

nat46v6Prefix

The prefix used for translating packets received from private IPv6 servers into IPv4 packets. This prefix has a length of 96 bits (128-32 = 96). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

nat46IgnoreTOS

Ignore TOS.

Possible values: YES, NO

Default value: NO

nat46ZeroCheckSum

Calculate checksum for UDP packets with zero checksum

Possible values: ENABLED, DISABLED

Default value: ENABLED

nat46v6Mtu

Calculate checksum for UDP packets with zero checksum

Default value: 1280

Minimum value: 1280

Maximum value: 1500

nat46FragHeader

When disabled, translator will not insert IPv6 fragmentation header for non fragmented IPv4 packets

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

set inat parameter -nat46ignoretos YES

---

unset inatparam

Use this command to remove inatparam settings.Refer to the set inatparam command for meanings of the arguments.

Synopsys

unset inatparam -nat46v6Prefix

---

show inatparam

Show the inat parameters.

Synopsys

show inatparam

Arguments

format

level

Outputs

nat46v6Prefix

The prefix used for translating packets received from private IPv6 servers into IPv4 packets. This prefix has a length of 96 bits (128-32 = 96). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

nat46IgnoreTOS

Ignore TOS.

nat46ZeroCheckSum

Calculate checksum for UDP packets with zero checksum

nat46v6Mtu

Calculate checksum for UDP packets with zero checksum

nat46FragHeader

When disabled, translator will not insert IPv6 fragmentation header for non fragmented IPv4 packets

Example

show inat params

inatsession

Sep 22, 2015

The following operations can be performed on "inatsession":

stat inatsession

Display statistics for stateful inat sessions.

Synopsys

stat inatsession <name> [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

INAT name

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

INAT total sessions (inatTotHits)

INAT total sessions

INAT Current sessions (inatCurSessions)

INAT current sessions

INAT total Received Bytes (inatTotReceiveBytes)

INAT total Received Bytes

INAT total Sent Bytes (inatTotSentBytes)

INAT total Sent Bytes

INAT total Packets Received (inatTotpktreceived)

INAT total Packets Received

INAT total Packets Sent (inatTotpktsent)

INAT total Packets Sent

Example

```
stat inatsession inat_1
```

interface

Sep 22, 2015

The following operations can be performed on "interface":

clear | set | unset | enable | disable | reset | show | stat

clear interface

Resets the statistical counters of the specified interface.

Synopsys

clear interface <id>@

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

* LA - Indicates a link aggregation port.

* LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

---

set interface

Modifies the parameters of an interface.

Synopsys

set interface <id>@ [-speed <speed>] [-duplex <duplex>] [-flowControl <flowControl>] [-autoneg (DISABLED | ENABLED)] [-haMonitor (ON | OFF)] [-tagall (ON | OFF)] [-lacpMode <lacpMode>] [-lacpKey <positive_integer>] [-lagtype (NODE | CLUSTER)] [-lacpPriority <positive_integer>] [-lacpTimeout (LONG | SHORT)] [-ifAlias <string>] [-throughput <positive_integer>] [-bandwidthHigh <positive_integer> [-bandwidthNormal <positive_integer>]]

Arguments

id

ID of the Interface whose parameters you want to modify.

For a NetScaler appliance, specify the interface in C/U notation (for example, 1/3).

For a cluster configuration, specify the interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

speed

Ethernet speed of the interface, in Mbps.

Notes:

* If you set the speed as AUTO, the NetScaler appliance attempts to auto-negotiate or auto-sense the link speed of the interface when it is UP. You must enable auto negotiation on the interface.

* If you set a speed other than AUTO, you must specify the same speed for the peer network device. Mismatched speed and duplex settings between the peer devices of a link lead to link errors, packet loss, and other errors.

Some interfaces do not support certain speeds. If you specify an unsupported speed, an error message appears.

Possible values: AUTO, 10, 100, 1000, 10000, 40000

Default value: NSA_DVC_SPEED_AUTO

duplex

The duplex mode for the interface.

Notes:

* If you set the duplex mode to AUTO, the NetScaler appliance attempts to auto-negotiate the duplex mode of the interface when it is UP. You must enable auto negotiation on the interface.

If you set a duplex mode other than AUTO, you must specify the same duplex mode for the peer network device. Mismatched speed and duplex settings between the peer devices of a link lead to link errors, packet loss, and other errors.

Possible values: AUTO, HALF, FULL

Default value: NSA_DVC_DUPLEX_AUTO

flowControl

802.3x flow control setting for the interface. The 802.3x specification does not define flow control for 10 Mbps and 100 Mbps speeds, but if a Gigabit Ethernet interface operates at those speeds, the flow control settings can be applied. The flow control setting that is finally applied to an interface depends on auto-negotiation. With the ON option, the peer negotiates the flow control, but the appliance then forces two-way flow control for the interface.

Possible values: OFF, RX, TX, RXTX

Default value: NSA_DVC_FC_OFF

autoneg

Auto-negotiation state of the interface.

With the ENABLED setting, the NetScaler appliance auto-negotiates the speed and duplex settings with the peer network device on the link.

The NetScaler appliance auto-negotiates the settings of only those parameters (speed or duplex mode) for which the value is set as AUTO.

Possible values: DISABLED, ENABLED

Default value: NSA_DVC_AUTONEG_ON

haMonitor

In a High Availability (HA) configuration, monitor the interface for failure events. In an HA configuration, an interface that has HA MON enabled and is not bound to any Failover Interface Set (FIS), is a critical interface. Failure or disabling of any critical interface triggers HA failover.

Possible values: ON, OFF

Default value: NSA_DVC_MONITOR_ON

tagall

Add a four-byte 802.1q tag to every packet sent on this interface. The ON setting applies the tag for this interface's native VLAN. OFF applies the tag for all VLANs other than the native VLAN.

Possible values: ON, OFF

Default value: NSA_DVC_VTRUNK_OFF

trunk

This argument is deprecated by tagall.

Possible values: ON, OFF

Default value: NSA_DVC_VTRUNK_OFF

lacpMode

Bind the interface to a LA channel created by the Link Aggregation control protocol (LACP).

Available settings function as follows:

* Active - The LA channel port of the NetScaler appliance generates LACPDU messages on a regular basis, regardless of any need expressed by its peer device to receive them.

* Passive - The LA channel port of the NetScaler appliance does not transmit LACPDU messages unless the peer device port is in the active mode. That is, the port does not speak unless spoken to.

* Disabled - Unbinds the interface from the LA channel. If this is the only interface in the LA channel, the LA channel is removed.

Possible values: DISABLED, ACTIVE, PASSIVE

Default value: NSA_LACP_DISABLE

lacpKey

Integer identifying the LACP LA channel to which the interface is to be bound.

For an LA channel of the NetScaler appliance, this digit specifies the variable x of an LA channel in LA/x notation, where x can range from 1 to 8. For example, if you specify 3 as the LACP key for an LA channel, the interface is bound to the LA channel LA/3.

For an LA channel of a cluster configuration, this digit specifies the variable y of a cluster LA channel in CLA/(y-4) notation, where y can range from 5 to 8. For example, if you specify 6 as the LACP key for a cluster LA channel, the interface is bound to the cluster LA channel CLA/2.

Minimum value: 1

Maximum value: 8

lagtype

Type of entity (NetScaler appliance or cluster configuration) for which to create the channel.

Possible values: NODE, CLUSTER

Default value: NSA_LAG_NODE

lacpPriority

LACP port priority, expressed as an integer. The lower the number, the higher the priority. The NetScaler appliance limits the number of interfaces in an LA channel to eight. If LACP is enabled on more than eight interfaces, the appliance selects eight interfaces, in descending order of port priority, to form a channel.

Default value: 32768

Minimum value: 1

Maximum value: 65535

lacpTimeout

Interval at which the NetScaler appliance sends LACPDU messages to the peer device on the LA channel.

Available settings function as follows:

LONG - 30 seconds.

SHORT - 1 second.

Possible values: LONG, SHORT

Default value: NSA_LACP_TIMEOUT_LONG

ifAlias

Alias name for the interface. Used only to enhance readability. To perform any operations, you have to specify the interface ID.

Default value: " "

throughput

Low threshold value for the throughput of the interface, in Mbps. In an HA configuration, failover is triggered if the interface has HA MON enabled and the throughput is below the specified the threshold.

Maximum value: 80000

bandwidthHigh

High threshold value for the bandwidth usage of the interface, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the interface is greater than or equal to the specified high threshold value.

Maximum value: 80000

---

unset interface

Use this command to remove interface settings.Refer to the set interface command for meanings of the arguments.

Synopsys

unset interface <id>@ [-speed] [-duplex] [-flowControl] [-autoneg] [-haMonitor] [-tagall] [-lacpMode] [-lacpKey] [-lacpPriority] [-lacpTimeout] [-ifAlias] [-throughput] [-bandwidthHigh] [-bandwidthNormal]

---

enable interface

Enables the interface. If the link is active, it can transmit and receive packets. Note: To view the status of an interface, use the show interface command.

Synopsys

enable interface <id>@

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

* LA - Indicates a link aggregation port.

* LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

---

disable interface

Disables the interface from transmitting and receiving packets. The link remains active and the peer network device is unaware that the interface has been disabled. In a High Availability configuration, an interface that has HA MON enabled and is not bound to any Failover Interface Set (FIS), is a critical interface. Disabling or failure of any critical interface triggers HA failover. Note: To view the status of an interface, use the show interface command.

Synopsys

disable interface <id>@

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

* LA - Indicates a link aggregation port.

* LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

---

reset interface

Restarts the interface but leaves the administrative state ENABLED or DISABLED and configuration unchanged. The link pertaining to the interface is reestablished with the existing settings.

Synopsys

reset interface <id>@

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

* LA - Indicates a link aggregation port.

* LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

---

show interface

Displays the settings of all interfaces or of the specified interface on the NetScaler appliance. To display the settings of all interfaces, run the command without any parameters. To display the settings of a particular interface, specify the ID of the interface.

Synopsys

show interface [<id>@] show interface stats - alias for 'stat interface'

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

* LA - Indicates a link aggregation port.

* LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

summary

fullValues

format

level

Outputs

stateflag

deviceName

Name of the interface.

unit

Unit number for this interface, signifying the sequence number in which this interface is discovered on this Netscaler.

description

Display the type of interface, the speeds at which this interface can operate, and, if applicable, the type of SFP,.

flags

Flags for this interface. Used for communicating the device states.

mtu

MTU for this interface (the largest frame that can transit this interface).

vlan

Native VLAN for this interface.

mac

MAC address for this interface.

uptime

Duration for which the interface has been UP (Example: 3 hours 1 minute 1 second). This value is reset when the interface state changes to DOWN..

downTime

Duration for which the interface has been DOWN. This value is reset when the interface state changes to UP.(Example: 3 hours 1 minute 1 second).

reqMedia

Requested media setting for this interface.

reqSpeed

Requested speed setting for this interface.

reqDuplex

Requested duplex setting for this interface.

reqFlowcontrol

Requested flow control setting for this interface.

media

Actual media setting for this interface.

speed

Actual speed setting for this interface.

duplex

Actual duplex setting for this interface.

flowControl

Actual flow control setting for this interface.

connDistr

Connection distribution setting on this interface.

macdistr

MAC distribution setting on this interface.

Mode

The mode(AUTO/MANNUAL) for the LA channel.

haMonitor

HA monitor enabled or disabled for this interface.

state

Link state of the interface (UP/DOWN).

autoneg

Interface autonegotiation enabled or disabled.

autonegResult

Actual auto-negotiation setting for this interface.

tagged

VLAN tags setting on this channel.

tagall

VLAN tagging behavior on this interface. With the ON setting,, packets are tagged with all the VLANs that are bound to this interface. With the OFF setting, packets are tagged with the native VLAN.

trunk

This argument is deprecated by tagall.NOTE: This attribute is deprecated.The "trunk" argument is confused with LA-trunk, renaming this to "tagall" instead.

taggedAny

Interface setting to accept/drop all tagged packets.

taggedAutolearn

Dynamic VLAN membership autolearning enabled or disabled on this interface.

hangDetect

Hang detection enabled or disabled for this interface.

hangReset

Hang reset enabled or disabled for this interface.

linkState

The current state of the link associated with the interface. For logical interfaces (LA), the state of the link is dependent on the state of the slave interfaces. For the link to be UP at least one of the slave interfaces needs to be UP.

intfState

Current state of the specified interface. The interface state set to UP only if the link state is UP and administrative state is ENABLED.

rxpackets

Number of packets received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

rxbytes

Number of bytes received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

rxerrors

Number of inbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were cleared. Packets can be dropped because of CRC, length (undersize or oversize), or alignment errors.

rxdrops

Number of inbound packets dropped by the specified interface. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler appliance when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. In most healthy networks, this statistic increments at a steady rate regardless of traffic load. A sharp spike in dropped packets generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

txpackets

Number of packets transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

txbytes

Number of bytes transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

txerrors

Number of outbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were

cleared. Packets can be dropped because of length (undersize or oversize) errors or a lack of resources. This statistic is available only for:

(1) Loop back interface (LO) of all platforms.

(2) All data ports on the NetScaler 12000 platform.

(3) Management ports on the Netscaler MPX 15000 and 17000 platforms.

txdrops

Number of packets dropped in transmission by the specified interface for one of the following reasons.

(1) VLAN mismatch.

(2) Oversized packets.

(3) Interface congestion.

(4) Loopback packets sent on non loop back interface.

inDisc

Number of error-free inbound packets discarded by the specified interface because of a lack of resources (for example, insufficient receive buffers).

outDisc

Number of error-free outbound packets discarded by the specified interface because of a lack of resources. This statistic is not available on:

(1) 10G ports of NetScaler MPX 12500/12500/15500-10G platforms.

(2) 10G data ports on NetScaler MPX 17500/19500/21500 platforms.

fctls

Number of times flow control is performed on the specified interface because of received pause frames.

hangs

Number of times the specified interface detected hangs in the transmit and receive paths since the NetScaler appliance was started or the interface statistics were cleared.

stsStalls

Number of times the status updates for a specified interface were stalled since the NetScaler appliance was started or the interface statistics were cleared. A status stall is detected when the status of the interface is not updated by the NIC hardware within 0.8 seconds of the last update.

txStalls

Number of times the interface stalled, when transmitting packets, since the NetScaler appliance was started or the interface statistics were cleared. Transmit (Tx) stalls are detected when a packet posted for transmission is not transmitted in 4 seconds.

rxStalls

Number of times the interface stalled, when receiving packets, since the NetScaler appliance was started or the interface statistics were cleared. Receive (Rx) stalls are detected when the following conditions are met:

(1)The link is up for more than 10 minutes.

(2)Packets are transmitted, but no packets are received for 16 seconds.

bdgMacMoved

Number of MAC moves between ports. A high rate of MAC moves typically indicates a bridge loop between two interfaces.

bdgMuted

Number of times the specified interface stopped transmitting and receiving packets because of MAC moves between ports.

vmac

Virtual MAC of this interface.

vmac6

Virtual MAC for IPv6 of this interface.

lacpMode

The LACP mode of the specified interface. The possible values are:

1. Active: A port in active mode generates LACP protocol messages on a regular basis, regardless of any need expressed by its partner to receive them.

2. Passive: A port in passive mode is generally not transmit LACP messages unless its partner is in the active mode; that is, it does not communicate to the other appliance unless other appliance communicates with this appliance.

lacpKey

Identifies the channel to which the interface is bound. The possible values are 1, 2, 3, and 4.

lacpPriority

LACP port priority, expressed as an integer. The lower the number, the higher the priority. The NetScaler appliance limits the number of interfaces in an LA channel to eight. If LACP is enabled on more than eight interfaces, the appliance selects eight interfaces, in descending order of port priority, to form a channel.

lacpTimeout

Time to wait for the LACPDU. If an LACPDU is not received within this interval, the NetScaler markes the link partner port as DOWN. Possible values; Long, Short. Long lacptimeout is 90 sec and Short LACP timeout is 3 sec.

lagtype

Type of entity (NetScaler appliance or cluster configuration) for which to create the channel.

ifAlias

Alias name for the interface. Used only to enhance readability. To perform any operations, you have to specify the interface ID.

reqThroughput

Minimum required throughput for an interface. Failover is triggered if the operating throughput of a Link Aggregation (LA) channel for which HAMON is ON falls below this value. The possible values are:

1. 1000Mbps for 1G interfaces.

2. 10000Mbps for 10G interfaces.

3. 80000Mbps for Link Aggregation channels.

throughput

Actual throughput for the interface.

bandwidthHigh

High threshold value for the bandwidth usage of the interface, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the interface is greater than or equal to the specified high threshold value.

bandwidthNormal

Normal threshold value for the bandwidth usage of the interface, in Mbps. When the bandwidth usage of the interface becomes less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

backplane

The cluster backplane status of the interface. If the status is enabled, the interface is part of the cluster backplane. By default, the backplane status is disabled.

ifnum

Contains the LA Master, if the interface is part of LA channel.

clearTime

Time since the interface stats are cleared last time.

slavestate

State of the member interfaces.

slavemedia

Media type of the member interfaces.

slavespeed

Speed of the member interfaces.

slaveduplex

Duplex of the member interfaces.

slaveflowctl

Flowcontrol of the member interfaces.

slavetime

UP time of the member interfaces.

intftype

Interface Type, this field will have the interface type either it is virtual, physical or loopback.

lacpActorMode

* Active - The LA channel port of the NetScaler appliance generates LACPDU messages on a regular basis, regardless of any need expressed by its peer

device to receive them.

* Passive - The LA channel port of the NetScaler appliance does not transmit LACPDU messages unless the peer device port is in the active mode. That is, the port does not speak unless spoken to.

* Disabled - Unbinds the interface from the LA channel. If this is the only interface in the LA channel, the LA channel is removed.

lacpActorTimeout

Interval at which the NetScaler appliance sends LACPDU messages to the peer device on the LA channel.

Available settings function as follows:

LONG - 30 seconds.

SHORT - 1 second.

lacpActorPriority

LACP Actor Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier. The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lacpActorPortno

LACP Actor port number. LACP uses the port priority with the port number to form the port identifier.

lacpPartnerState

LACP Partner State. Whether the port is in Active or Passive negotiating state.

lacpPartnerTimeout

The timeout value for the information revieved in LACPDUs. It can have values as SHORT or LONG. The SHORT timeout is 3s and the LONG timeout is 90s.

lacpPartnerAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lacpPartnerInsync

The Synchronization flag indicates that the transmitting participant.s mux component is in sync with the system id and key information transmitted.

lacpPartnerCollecting

The Collecting flag indicates that the participant.s collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lacpPartnerDistributing

The Distributing flag indicates that the participant.s distributor is not definitely off. If reset the flag indicates not distributing.

lacpPartnerDefaulted

If the timer expires in the Expired state, the Receive Machine enters the Defaulted state.

lacpPartnerExpired

If the LACPDUs are received for timeout period, the Receive Machine enters the Expired state and the timer is restarted with the timeout value of SHORT timeout

lacpPartnerPriority

LACP Partner Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier.

The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lacpPartnerSystemMac

LACP Partner System MAC.

lacpPartnerSystemPriority

LACP Partner System Priority. The LACP partner's system priority. The values for the priority range from 0 to 65535. The lower the value, the higher the system priority. The switch with the lower system priority value determines which links between LACP partner are active and which are in the standby for each LACP Channel.

lacpPartnerPortno

LACP Partner Port number. LACP uses the port priority with the port number to form the port identifier.

lacpPartnerKey

LACP Partner Key. The LACP key used by the partner port.

lacpActorAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lacpActorInsync

The Synchronization flag indicates that the transmitting participant.s mux component is in sync with the system id and key information transmitted.

lacpActorCollecting

The Collecting flag indicates that the participant.s collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lacpActorDistributing

The Distributing flag indicates that the participant.s distributor is not definitely off. If reset the flag indicates not distributing.

lacpPortMuxState

LACP Port MUX state. The state of the MUX control machine. The Mux Control Machine attaches the physical port to an aggregate port, using the Selection Logic to choose an appropriate port, and turns the distributor and collector for the physical port on or off as required by protocol information.

lacpPortRxStat

LACP Port RX state. The state of the Receive machine. The Receive Machine maintains partner information, recording protocol information from LACPDUs sent by remote partner(s). Received information is subject to a timeout, and if sufficient time elapses the receive machine will revert to using default partner information.

lacpPortSelectState

LACP Port SELECT state. The state of the SELECT state machine, It could be SELECTED or UNSELECTED.

devno

count

Example

The output for the show interface command is as follows: 1) Interface 0/1 (Gig Ethernet 10/100/1000 MBits) #4 flags=0x4021

---

stat interface

Displays the statistics of all interfaces or of the specified interface on the NetScaler appliance. To display the statistics of all interfaces, run the command without any parameters. To display the statistics of a particular interface, specify the ID of the interface.

Synopsys

stat interface [<id>@] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

* LA - Indicates a link aggregation port.

* LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Interface State (IntfState)

Current state of the specified interface. The interface state set to UP only if the link state is UP and administrative state is ENABLED .

Link uptime (UpTime)

Duration for which the link is UP. This statistic is reset when the state changes to DOWN.

Link downtime (DnTime)

Duration for which the link is DOWN. This statistic is reset when the state changes to UP.

Bytes received (Rx Bytes)

Number of bytes received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Bytes transmitted (Tx Bytes)

Number of bytes transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Packets received (Rx Pkts)

Number of packets received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Packets transmitted (Tx Pkts)

Number of packets transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Multicast packets (McastPkt)

Number of multicast packets received by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

NetScaler packets (NSPkt)

Number of packets, destined to the NetScaler, received by an interface since the NetScaler appliance was started or the interface statistics were cleared. The packets destined to NetScaler are those that have the same MAC address as that of an interface or a VMAC address owned by the NetScaler.

LACPDUs received (RxLacpdu)

Number of Link Aggregation Control Protocol Data Units(LACPDUs) received by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

LACPDUs transmitted (TxLacpdu)

Number of Link Aggregation Control Protocol Data Units(LACPDUs) transmitted by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

Error packets received (hw) (ErrRx)

Number of inbound packets dropped by the hardware on a specified interface once the NetScaler appliance starts or the interface statistics are cleared. This happens due to following reasons:

1) The hardware receives packets at a rate higher rate than that at which the software is processing packets. In this case, the hardware FIFO overruns and starts dropping the packets .

2) The specified interface fails to receive inbound packets from the appliance because of insufficient memory.

3) The specified interface receives packets with CRC errors (Alignment or Frame Check Sequence).

4) The specified interface receives overly long packets.

5) The specified interface receives packets with alignment errors.

6) The software does less buffering because it is running out of available memory. When hardware detects that there is no space into which to push newly arrived packets, it starts dropping them.

7) The specified interface receives packets with Frame Check Sequence (FCS) errors.

8) The specified interface receives packets smaller than 64 bytes.

9) The specified interface discards error-free inbound packets because of insufficient resources. For example: NIC buffers.

10) Packets are missed because of collision detection, link lost, physical decoding error, or MAC abort.

Error packets transmitted (hw) (ErrTx)

Number of outbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were cleared. This could happen due to length (undersize or oversize) errors and lack of resources. This statistic is available only for:

(1) Loop back interface (LO) of all platforms.

(2) All data ports on the NetScaler 12000 platform.

(3) Management ports on the MPX 15000 and 17000 platforms.

Inbound packets discarded(hw) (InDisc)

Number of error-free inbound packets discarded by the specified interface due to a lack of resources, for example, insufficient receive buffers.

Outbound packets discarded(hw) (OutDisc)

Number of error-free outbound packets discarded by the specified interface due to a lack of resources. This statistic is not available on:

(1) 10G ports of NetScaler MPX 12500/12500/15500-10G platforms.

(2) 10G data ports on NetScaler MPX 17500/19500/21500 platforms.

Packets dropped in Rx (sw) (DrpRxPkt)

Number of inbound packets dropped by the specified interface. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. This statistic will increment in most healthy networks at a steady rate regardless of traffic load. If a sharp spike in dropped packets occurs, it generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

Packets dropped in Tx (sw) (DrpTxPkt)

Number of packets dropped in transmission by the specified interface due to one of the following reasons.

(1) VLAN mismatch.

(2) Oversized packets.

(3) Interface congestion.

(4) Loopback packets sent on non loop back interface.

NIC hangs (Hangs)

Number of times the specified interface detected hangs in the transmit and receive paths since the NetScaler appliance was started or the interface statistics were cleared.

Status stalls (StsStall)

Number of times the status updates for a specified interface were stalled since the NetScaler appliance was started or the interface statistics were cleared. A status stall is detected when the status of the interface is not updated by the NIC hardware within 0.8 seconds of the last update.

Transmit stalls (TxStall)

Number of times the interface stalled, when transmitting packets, since the NetScaler appliance was started or the interface statistics were cleared. Transmit (Tx) stalls are detected when a packet posted for transmission is not transmitted in 4 seconds.

Receive stalls (RxStall)

Number of times the interface stalled, when receiving packets, since the NetScaler appliance was started or the interface statistics were cleared. Receive (Rx) stalls are detected when the following conditions are met:

(1)The link is up for more than 10 minutes.

(2)Packets are transmitted, but no packets is received for 16 seconds.

Error-disables (ErrDis)

Number of times the specified interface is disabled by the NetScaler, due to continuous Receive (Rx) or Transmit (Tx) stalls, since the NetScaler appliance was started or the interface statistics were cleared. The NetScaler disables an interface when one of the following conditions is met:

(1) Three consecutive transmit stalls occurs with at most gap of 10 seconds between any two stalls.

(2) Three consecutive receive stalls occurs with at most gap of 120 seconds between any two stalls.

Duplex mismatches (DupMism)

Number of times duplex mismatches were detected on the specified interface since the NetScaler appliance was started or the interface statistics were cleared. A mismatch will occur if the duplex mode is not identically set on both ends of the link. This statistic is only available on the NetScaler Classic edition.

Link re-initializations (LnkReint)

Number of times the link has been re-initialized. A re-initialization occurs due to link state change, configuration parameter change, or administrative reset operation.

MAC moves registered (MacMvd)

Number of MAC moves between ports. If a high rate of MAC moves is observed, it is likely that there is a bridge loop between two interfaces.

Times NIC became muted (ErrMtd)

Number of times the specified interface stopped transmitting and receiving packets due to MAC moves between ports.

Interface Alias (IntfAlias)

Alias Name for the Interface

Link State (State)

The current state of the link associated with the interface. For logical interfaces (LA), the state of the link is dependent on the state of the slave interfaces. For the link to be UP at least one of the slave interfaces needs to be UP.

ip6Tunnel

Sep 22, 2015

The following operations can be performed on "ip6Tunnel":

add | rm | show

add ip6Tunnel

Creates an IPv6 tunnel. An IP tunnel is a communication channel, using encapsulation technologies, between two networks that do not have a routing path. Every IP packet that is shared between the two networks is encapsulated within another packet and then sent through the tunnel.

Synopsys

add ip6Tunnel <name> <remote> <local>

Arguments

name

Name for the IPv6 Tunnel. Cannot be changed after the service group is created. Must begin with a number or letter, and can consist of letters, numbers, and the @ _ - . (period) : (colon) # and space ( ) characters.

remote

An IPv6 address of the remote NetScaler appliance used to set up the tunnel.

local

An IPv6 address of the local NetScaler appliance used to set up the tunnel.

Example

```
add ip6tunnel tun6 9901::200/64 *
```

---

rm ip6Tunnel

Removes an IPv6 tunnel from the NetScaler appliance.

Synopsys

rm ip6Tunnel <name>

Arguments

name

Name of the IPv6 tunnel to be removed.

Example

```
rm ip6tunnel tun6
```

---

show ip6Tunnel

Displays the settings of all IPv6 tunnels configured on the NetScaler appliance, or of the specified IPv6 tunnel.

Synopsys

show ip6Tunnel [<name> | <remote>]

Arguments

name

Name of the IPv6 tunnel whose details you want to display.

remote

The IPv6 address at which the remote NetScaler appliance connects to the tunnel.

summary

fullValues

format

level

Outputs

remoteIP

The remote IP address or subnet of the tunnel.

local

An IPv6 address of the local NetScaler appliance used to set up the tunnel.

type

The type of this tunnel.

encapIp

The effective local IP address of the tunnel. Used as the source of the encapsulated packets.

devno

count

stateflag

Example

```
1) Name.....: tun61    Remote.....:    9901::200/64 Local.....:        *    Encap.....:      ::0/128     Type.....:          C 2) Na
```

ip6TunnelParam

Sep 22, 2015

The following operations can be performed on "ip6TunnelParam":

set | unset | show

set ip6TunnelParam

Sets global parameters of IPv6 tunnels on the NetScaler appliance.

Synopsys

set ip6TunnelParam [-srcIP <ipv6_addr|null>] [-dropFrag (YES | NO)] [-dropFragCpuThreshold <positive_integer>] [-srcIPRoundRobin (YES | NO)]

Arguments

srcIP

Common source IPv6 address for all IPv6 tunnels. Must be a SNIP6 or VIP6 address.

dropFrag

Drop any packet that requires fragmentation.

Possible values: YES, NO

Default value: NO

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation. Applies only if dropFragparameter is set to NO.

Minimum value: 1

Maximum value: 100

srcIPRoundRobin

Use a different source IPv6 address for each new session through a particular IPv6 tunnel, as determined by round robin selection of one of the SNIP6 addresses. This setting is ignored if a common global source IPv6 address has been specified for all the IPv6 tunnels. This setting does not apply to a tunnel for which a source IPv6 address has been specified.

Possible values: YES, NO

Default value: NO

Example

```
set ip6TunnelParam -srcIP 9901::100 -dropFrag YES -dropFragCpuThreshold 95
```

---

unset ip6TunnelParam

Resets the specified global parameters of IPv6 tunnels to their default settings. Refer to the set ip6TunnelParam command for parameter descriptions..Refer to the set ip6TunnelParam command for meanings of the arguments.

Synopsys

unset ip6TunnelParam [-srcIP] [-dropFrag] [-dropFragCpuThreshold] [-srcIPRoundRobin]

Example

unset ip6TunnelParam -srcIP -dropFrag -dropFragCpuThreshold

---

show ip6TunnelParam

Displays the global settings of IPv6 tunnels on the NetScaler appliance.

Synopsys

show ip6TunnelParam

Arguments

format

level

Outputs

srcIP

Common source IPv6 address for all IPv6 tunnels. Must be a SNIP6 or VIP6 address.

dropFrag

Drop any packet that requires fragmentation.

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation. Applies only if dropFragparameter is set to NO.

srcIPRoundRobin

Use a different source IPv6 address for each new session through a particular IPv6 tunnel, as determined by round robin selection of one of the SNIP6 addresses. This setting is ignored if a common global source IPv6 address has been specified for all the IPv6 tunnels. This setting does not apply to a tunnel for which a source IPv6 address has been specified.

Example

Tunnel Source IP: 9901::100 Drop if Fragmentation Needed: YES CPU usage threshold to avoid fragmentation: 95

---

ipTunnel

Sep 22, 2015

The following operations can be performed on "ipTunnel":

add | rm | show

add ipTunnel

Creates an IPv4 tunnel. An IP tunnel is a communication channel, using encapsulation technologies, between two networks that do not have a routing path. Every IP packet that is shared between the two networks is encapsulated within another packet and then sent through the tunnel.

Synopsys

add ipTunnel <name> <remote> <remoteSubnetMask> <local> [-protocol <protocol>] [-ipsecProfileName <string>]

Arguments

name

Name for the IP tunnel. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ( ).

remote

Public IPv4 address, of the remote device, used to set up the tunnel. For this parameter, you can alternativelyspecify a network address if you specify IPIP (IP over IP) for the Protocol parameter.

remoteSubnetMask

Subnet mask of the remote IP address of the tunnel.

local

Type ofNetScaler owned public IPv4 address, configured on the local NetScaler appliance and used to set up the tunnel.

protocol

The IP tunneling protocol.

Possible values: IPIP, GRE, IPSEC

Default value: TNL_IPIP

ipsecProfileName

Name of IPSec profile to be associated.

Default value: "ns_ipsec_default_profile"

Example

```
add iptunnel tunnel1 10.100.20.0 255.255.255.0 *
```

---

rm ipTunnel

Removes an IP tunnel configuration from the NetScaler appliance.

Synopsys

rm ipTunnel <name>

Arguments

name

Name of the IP Tunnel.

Example

```
rm iptunnel tunnel1
```

---

show ipTunnel

Display the configured IP tunnels.

Synopsys

show ipTunnel [(<remote> <remoteSubnetMask>) | <name>]

Arguments

remote

Public IPv4 address, of the remote device, used to set up the tunnel. For this parameter, you can alternativelyspecify a network address if you specify IPIP (IP over IP) for the Protocol

parameter.

name

Name for the IP tunnel. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ( ).

summary

fullValues

format

level

Outputs

name

Name for the PBR

local

Type of NetScaler owned public IPv4 address, configured on the local NetScaler appliance and used to set up the tunnel.

protocol

The IP tunneling protocol.

type

The type of this tunnel.

encapIp

The effective local IP address of the tunnel. Used as the source of the encapsulated packets.

channel

The tunnel that is bound to a netbridge.

ipsecProfileName

Name of IPSec profile to be associated.

tunnelType

Indicates that a tunnel is User-Configured, Internal or DELETE-IN-PROGRESS.

ipsecTunnelStatus

Whether the ipsec on this tunnel is up or down.

devno

count

stateflag

Example

```
1) Name.....: t1    Remote.....:    10.102.33.0  Mask.....:  255.255.255.0    Local.....:          *  Encap.....:       0.0.0.0    P
```

ipTunnelParam

Sep 22, 2015

The following operations can be performed on "ipTunnelParam":

set | unset | show

set ipTunnelParam

Sets global parameters of IPv4 tunnels on the NetScaler appliance.

Synopsys

set ipTunnelParam [-srcIP <ip_addr>] [-dropFrag (YES | NO)] [-dropFragCpuThreshold <positive_integer>] [-srcIPRoundRobin (YES | NO)]

Arguments

srcIP

Common source-IP address for all tunnels. For a specific tunnel, this global setting is overridden if you have specified another source IP address. Must be a MIP or SNIP address.

dropFrag

Drop any IP packet that requires fragmentation before it is sent through the tunnel.

Possible values: YES, NO

Default value: NO

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation to use the IP tunnel. Applies only if dropFragparameter is set to NO. The default value, 0, specifies that this parameter is not set.

Minimum value: 1

Maximum value: 100

srcIPRoundRobin

Use a different source IP address for each new session through a particular IP tunnel, as determined by round robin selection of one of the SNIP addresses. This setting is ignored if a common global source IP address has been specified for all the IP tunnels. This setting does not apply to a tunnel for which a source IP address has been specified.

Possible values: YES, NO

Default value: NO

Example

```
set ipTunnelParam -srcIP 10.100.20.48 -dropFrag YES -dropFragCpuThreshold 95
```

---

unset ipTunnelParam

Use this command to remove ipTunnelParam settings.Refer to the set ipTunnelParam command for meanings of the arguments.

Synopsys

unset ipTunnelParam [-srcIP] [-dropFrag] [-dropFragCpuThreshold] [-srcIPRoundRobin]

---

show ipTunnelParam

Display the IP Tunnel global settings on the NetScaler

Synopsys

show ipTunnelParam

Arguments

format

level

Outputs

srcIP

Common source-IP address for all tunnels. For a specific tunnel, this global setting is overridden if you have specified another source IP address. Must be a MIP or SNIP address.

dropFrag

Drop any IP packet that requires fragmentation before it is sent through the tunnel.

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation to use the IP tunnel. Applies only if dropFragparameter is set to NO. The default value, 0, specifies that this parameter is not set.

srcIPRoundRobin

Use a different source IP address for each new session through a particular IP tunnel, as determined by round robin selection of one of the SNIP addresses. This setting is ignored if a common global source IP address has been specified for all the IP tunnels. This setting does not apply to a tunnel for which a source IP address has been specified.

Example

```
Tunnel Source IP: 10.100.20.48 Drop if Fragmentation Needed: YES CPU usage threshold to avoid fragmentation: 95
```

ipset

Sep 22, 2015

The following operations can be performed on "ipset":

add | rm | bind | unbind | show

add ipset

Creates an IP set to which you can bind subnet IP (SNIP) or mapped IP (MIP) addresses that have been configured on the NetScaler appliance.

Synopsys

add ipset <name> [-td <positive_integer>]

Arguments

name

Name for the IP set. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the IP set is created. Choose a name that helps identify the IP set.

td

Traffic Domain Id.

Maximum value: 4094

Example

```
add ipset pool1
```

---

rm ipset

Removes an IP set from the NetScaler appliance.

Synopsys

rm ipset <name> ...

Arguments

name

Name of the IP set to be removed.

Example

```
rm ipset pool1
```

bind ipset

Binds specified IP addresses to an IP set.

Synopsys

bind ipset <name> <IPAddress>@ ...

Arguments

name

Name of the IP set to which to bind IP addresses.

IPAddress

SNIP or MIP addresses, configured on the NetScaler appliance, to be bound to the IP set. (If using the CLI, use spaces to separate multiple addresses.)

Example

```
bind ipset ipset_1 10.102.1.10
```

unbind ipset

Unbinds the associated IP addresses from an IP set.

Synopsys

unbind ipset <name> <IPAddress>@ ...

Arguments

name

Name of the IP set from which to unbind IP addresses.

IPAddress

IP addresses to be unbound from the IP set. (If using the CLI, use spaces to separate multiple addresses.)

Example

```
unbind ipset ipset_1 10.102.1.10
```

show ipset

Displays the settings of all IP sets configured on the NetScaler appliance, or of the specified IP set.

Synopsys

show ipset [<name>]

Arguments

name

Name of the IP set whose details you want to display.

summary

fullValues

format

level

Outputs

td

Traffic Domain Id.

IPAddress

One or more IP addresses bound to the IP set.

stateflag

state flag

flags

devno

count

Example

show network ipset

ipv6

Sep 22, 2015

The following operations can be performed on "ipv6":

set | unset | show

set ipv6

Sets the IPv6-related parameters: RA Learning and IPv6 NAT Prefix.

Synopsys

set ipv6 [-ralearning (ENABLED | DISABLED)] [-routerRedirection (ENABLED | DISABLED)] [-ndBasereachTime <positive_integer>] [-ndRetransmissionTime <positive_integer>] [-natprefix <ipv6_addr|*>] [-doDAD (ENABLED | DISABLED)]

Arguments

ralearning

Enable the NetScaler appliance to learn about various routes from Router Advertisement (RA) and Router Solicitation (RS) messages sent by the routers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

routerRedirection

Enable the NetScaler appliance to do Router Redirection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ndBasereachTime

ND6 base reachable time (ms)

Default value: 30000

ndRetransmissionTime

ND6 retransmission time (ms)

Default value: 1000

natprefix

Prefix used for translating packets from private IPv6 servers to IPv4 packets. This prefix has a length of 96 bits (128-32 = 96). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as

the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

doDAD

Enable the NetScaler appliance to do Duplicate Address Detection(DAD) for all the IPv6 addresses configured on NS, regardless of whether they are obtained through stateless autoconfiguration .DHCPv6, or manual configuration. RFC4862-sec 5.4.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set ipv6 -natprefix 2000::/96
```

---

unset ipv6

Use this command to remove ipv6 settings.Refer to the set ipv6 command for meanings of the arguments.

Synopsys

```
unset ipv6 [-ralearning] [-routerRedirection] [-ndBasereachTime] [-ndRetransmissionTime] [-natprefix] [-doDAD]
```

---

show ipv6

Display IPv6 settings

Synopsys

```
show ipv6
```

Arguments

format

level

Outputs

ralearning

Enable the NetScaler appliance to learn about various routes from Router Advertisement (RA) and Router Solicitation (RS) messages sent by the routers.

routerRedirection

Enable the NetScaler appliance to do Router Redirection.

basereachtime

ND6 base reachable time (ms)NOTE: This attribute is deprecated.depricating "basereachtime" in favor of "ndBasereachTime" to accommodate larger value set

ndBasereachTime

ND6 base reachable time (ms)

reachtime

ND6 computed reachable time (ms)NOTE: This attribute is deprecated.depricating V6REACHTIME in favor of ND6REACHTIME to accommodate larger value set

ndreachtime

ND6 computed reachable time (ms)

retransmissiontime

ND6 retransmission time (ms)NOTE: This attribute is deprecated.depricating "retransmissiontime" in favor of "ndRetransmissionTime" to accommodate larger value set

ndRetransmissionTime

ND6 retransmission time (ms)

natprefix

Prefix used for translating packets from private IPv6 servers to IPv4 packets. This prefix has a length of 96 bits (128-32 = 96). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

doDAD

Enable the NetScaler appliance to do Duplicate Address Detection(DAD) for all the IPv6 addresses configured on NS, regardless of whether they are obtained through stateless autoconfiguration .DHCPv6, or manual configuration. RFC4862-sec 5.4.

Example

show ipv6

lacp

Sep 22, 2015

The following operations can be performed on "lacp":

set | show

set lacp

Sets the Link Aggregation Control Protocol (LACP) system priority. Note: The NetScaler appliance automatically adds a parameter called mac in the configuration file (ns.conf) for this command entry. This parameter is set to the MAC address of one of the NetScaler appliance's interfaces and is used along with the system priority to form the system ID for the LACP channel.

Synopsys

set lacp -sysPriority <positive_integer> [-ownerNode <positive_integer>]

Arguments

sysPriority

Priority number that determines which peer device of an LACP LA channel can have control over the LA channel. This parameter is globally applied to all LACP channels on the NetScaler appliance. The lower the number, the higher the priority.

Default value: 32768

Minimum value: 1

Maximum value: 65535

ownerNode

The owner node in a cluster for which we want to set the lacp priority. Owner node can vary from 0 to 31. Ownernode value of 254 is used for Cluster.

Default value: 255

---

show lacp

Displays the settings of all channels created by the link aggregation control protocol (LACP) on the NetScaler appliance.

Synopsys

show lacp [-ownerNode <positive_integer>]

Arguments

ownerNode

The owner node in a cluster for which we want to set the lacp priority. Owner node can vary from 0 to 31. Ownernode value of 254 is used for Cluster.

Default value: 255

format

level

Outputs

deviceName

Name of the channel.

sysPriority

Priority number that determines which peer device of an LACP LA channel can have control over the LA channel. This parameter is globally applied to all LACP channels on the NetScaler appliance. The lower the number, the higher the priority.

mac

LACP system MAC.

flags

Flags of this channel.

lacpKey

LACP key of this channel.

clustersysPriority

LACP system (Cluster) priority

clusterMac

LACP system (Cluster) mac.

devno

count

stateflag

linkset

Sep 22, 2015

The following operations can be performed on "linkset":

add | rm | bind | unbind | show

add linkset

Adds a linkset to the NetScaler cluster.

Synopsys

add linkset <id>

Arguments

id

Unique identifier for the linkset. Must be of the form LS/x, where x can be an integer from 1 to 32.

Example

```
add linkset LS/1
```

---

rm linkset

Removes a linkset from the cluster.

Synopsys

rm linkset <id>

Arguments

id

ID of the linkset to be removed.

Example

```
rm linkset LS/1
```

---

bind linkset

Binds interfaces to the linkset.

Synopsys

bind linkset <id> -ifnum <interface_name> ...

Arguments

id

ID of the linkset to which to bind the interfaces.

ifnum

The interfaces to be bound to the linkset.

Example

```
bind linkset LS/1 -ifnum 1/1/1
```

---

unbind linkset

Unbinds interfaces from the linkset.

Synopsys

unbind linkset <id> -ifnum <interface_name> ...

Arguments

id

ID of the linkset from which to unbind the interfaces.

ifnum

Interfaces to be unbound from the linkset.

Example

```
unbind linkset LS/1 -ifnum 1/1/1
```

---

show linkset

Displays information about all linksets, or displays information about the specified linkset.

Synopsys

show linkset [<id>]

Arguments

id

ID of the linkset for which to display information. If an ID is not provided, the display includes information about all linksets that are available in the cluster.

summary

fullValues

format

level

Outputs

ifnum

The interfaces to be bound to the linkset.

stateflag

state flag

devno

count

Example

show linkset

nat64

Sep 22, 2015

The following operations can be performed on "nat64":

add | set | unset | rm | stat | show

add nat64

Configure a nat64 rule on the appliance.

Synopsys

add nat64 <name> <acl6name> [-netProfile <string>]

Arguments

name

Name of NAT64 rule.

acl6name

The ACL6 name.

netProfile

The name of the network profile.

---

set nat64

Set the configured nat64 rule.

Synopsys

set nat64 <name> [-acl6name <string>] [-netProfile <string>]

Arguments

name

Name of NAT64 rule.

acl6name

The ACL6 name.

netProfile

The name of the network profile.

Example

set nat64 rule1 -acl6name acl1 .

---

unset nat64

Use this command to remove nat64 settings.Refer to the set nat64 command for meanings of the arguments.

Synopsys

unset nat64 <name> -netProfile

---

rm nat64

Remove the configured nat64 rule.

Synopsys

rm nat64 <name>

Arguments

name

Name of NAT64 rule.

Example

rm nat64 name.

---

stat nat64

Display statistics for nat64 sessions.

Synopsys

stat nat64 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

TCP Sessions (nat64TotTcpSessions)

Total number of TCP sessions created by NAT64.

UDP Sessions (nat64TotUdpSessions)

Total number of UDP sessions created by NAT64.

ICMP Sessions (nat64TotIcmpSessions)

Total number of ICMP sessions created by NAT64.

Total Sessions (nat64TotSessions)

Total number of sessions created by NAT64.

Example

stat nat64

---

show nat64

Display the nat64 configuration.

Synopsys

show nat64 [<name>]

Arguments

name

Name of NAT64 rule.

format

level

Outputs

acl6name

The ACL6 name.

netProfile

The name of the network profile.

devno

count

stateflag

---

nd6

Sep 22, 2015

The following operations can be performed on "nd6":

add | clear | rm | show

add nd6

Adds a static entry to the ND6 table of the NetScaler appliance.

Synopsys

add nd6 <neighbor> <mac> <ifnum> [-vlan <integer>] [-td <positive_integer>]

Arguments

neighbor

Link-local IPv6 address of the adjacent network device to add to the ND6 table.

mac

MAC address of the adjacent network device.

ifnum

Interface through which the adjacent network device is available, specified in slot/port notation (for example, 1/3). Use spaces to separate multiple entries.

vlan

Integer value that uniquely identifies the VLAN on which the adjacent network device exists.

Minimum value: 1

Maximum value: 4094

td

Traffic Domain Id.

Maximum value: 4094

Example

```
add nd6 2001::1 00:04:23:be:3c:06 5 1/1
```

---

clear nd6

Removes all IPv6 neighbour discovery entries from the NetScaler appliance.

Synopsys

clear nd6

---

rm nd6

Remove a static IPv6 neighbor discovery entry from the NetScaler appliance's ND6 table.

Synopsys

rm nd6 <neighbor> [-vlan <integer>] [-td <positive_integer>]

Arguments

neighbor

Link-local IPv6 address of the adjacent network device that you want to remove from the ND6 table.

vlan

Integer value that uniquely identifies the VLAN for the ND6 entry you want to remove.

Minimum value: 1

Maximum value: 4094

td

Traffic Domain Id.

Maximum value: 4094

Example

rm nd6 2001::1 5 1/1

---

show nd6

Display the neighbor discovery information.

Synopsys

show nd6 [<neighbor> [-td <positive_integer>]]

Arguments

neighbor

Link-local IPv6 address of the adjacent network device to add to the ND6 table.

summary

fullValues

format

level

Outputs

mac

MAC address of the adjacent network device.

state

ND6 state

timeout

Time elapsed

ifnum

Interface through which the adjacent network device is available, specified in slot/port notation (for example, 1/3). Use spaces to separate multiple entries.

vlan

Integer value that uniquely identifies the VLAN on which the adjacent network device exists.

flags

flag for static/permanent entry.

channel

The tunnel that is bound to a netbridge.

devno

count

stateflag

Example

Following is an example of the output for the show nd6 command: Neighbor MAC-Address(Vlan, Interface) State TIME(hh:

---

nd6RAvariables

Sep 22, 2015

The following operations can be performed on "nd6RAvariables":

set | unset | show | bind | unbind

set nd6RAvariables

Set vlan specific Router Advertisment parameters in NetScaler.

Synopsys

set nd6RAvariables -vlan <positive_integer> [-ceaseRouterAdv (YES | NO)] [-sendRouterAdv (YES | NO)] [-srcLinkLayerAddrOption (YES | NO)] [-onlyUnicastRtAdvResponse (YES | NO)] [-managedAddrConfig (YES | NO)] [-otherAddrConfig (YES | NO)] [-currHopLimit <positive_integer>] [-maxRtAdvInterval <positive_integer>] [-minRtAdvInterval <positive_integer>] [-linkMTU <positive_integer>] [-reachableTime <positive_integer>] [-retransTime <positive_integer>] [-defaultLifeTime <integer>]

Arguments

vlan

The VLAN number.

Maximum value: 4094

ceaseRouterAdv

Cease router advertisements on this vlan.

Possible values: YES, NO

Default value: NO

sendRouterAdv

whether the router sends periodic RAs and responds to Router Solicitations.

Possible values: YES, NO

Default value: NO

srcLinkLayerAddrOption

Include source link layer address option in RA messages.

Possible values: YES, NO

Default value: YES

onlyUnicastRtAdvResponse

Send only Unicast Router Advertisements in respond to Router Solicitations.

Possible values: YES, NO

Default value: NO

managedAddrConfig

Value to be placed in the Managed address configuration flag field.

Possible values: YES, NO

Default value: NO

otherAddrConfig

Value to be placed in the Other configuration flag field.

Possible values: YES, NO

Default value: NO

currHopLimit

Current Hop limit.

Default value: 64

Maximum value: 255

maxRtAdvInterval

Maximum time allowed between unsolicited multicast RAs, in seconds.

Default value: 600

Minimum value: 4

Maximum value: 1800

minRtAdvInterval

Minimum time interval between RA messages, in seconds.

Default value: 198

Minimum value: 3

Maximum value: 1350

linkMTU

The Link MTU.

Maximum value: 1500

reachableTime

Reachable time, in milliseconds.

Maximum value: 3600000

retransTime

Retransmission time, in milliseconds.

defaultLifeTime

Default life time, in seconds.

Default value: 1800

Maximum value: 9000

Example

```
set nd6RAvariables -vlan 2 -maxRtAdvInterval 600
```

---

unset nd6RAvariables

Use this command to remove nd6RAvariables settings.Refer to the set nd6RAvariables command for meanings of the arguments.

Synopsys

```
unset nd6RAvariables -vlan <positive_integer> [-ceaseRouterAdv] [-sendRouterAdv] [-srcLinkLayerAddrOption] [-onlyUnicastRtAdvResponse] [-managedAddrConfig] [-otherAddrConfig] [-currHopLimit] [-maxRtAdvInterval] [-minRtAdvInterval] [-linkMTU] [-reachableTime] [-retransTime] [-defaultLifeTime]
```

---

show nd6RAvariables

Display Router Advertisement configuration variables.

Synopsys

```
show nd6RAvariables [-vlan <positive_integer>]
```

Arguments

vlan

The VLAN number.

Maximum value: 4094

format

level

Outputs

ceaseRouterAdv

Cease router advertisements on this vlan.

sendRouterAdv

whether the router sends periodic RAs and responds to Router Solicitations.

srcLinkLayerAddrOption

Include source link layer address option in RA messages.

onlyUnicastRtAdvResponse

Send only Unicast Router Advertisements in respond to Router Solicitations.

managedAddrConfig

Value to be placed in the Managed address configuration flag field.

otherAddrConfig

Value to be placed in the Other configuration flag field.

currHopLimit

Current Hop limit.

maxRtAdvInterval

Maximum time allowed between unsolicited multicast RAs, in seconds.

minRtAdvInterval

Minimum time interval between RA messages, in seconds.

linkMTU

The Link MTU.

reachableTime

Reachable time, in milliseconds.

retransTime

Retransmission time, in milliseconds.

defaultLifeTime

Default life time, in seconds.

stateflag

RA Param state flags.

lastRtAdvTime

Last RA sent timestamp.

nextRtAdvDelay

Next RA delay.

ipv6Prefix

Onlink prefixes for RA messages.

devno

count

---

bind nd6RAvariables

Bind on-link global prefixes to Router Advertisments variables.

Synopsys

bind nd6RAvariables -vlan <positive_integer> -ipv6Prefix <ipv6_addr|*>

Arguments

vlan

The VLAN number.

Maximum value: 4094

ipv6Prefix

Onlink prefixes for RA messages.

Example

```
bind nd6RAvariables -vlan 2 -ipv6Prefix 8000::/64
```

---

unbind nd6RAvariables

Unbind prefix from Router Advertisment parameters in NetScaler

Synopsys

unbind nd6RAvariables -vlan <positive_integer> -ipv6Prefix <ipv6_addr|*>

Arguments

vlan

The VLAN number.

Maximum value: 4094

ipv6Prefix

Onlink prefixes for RA messages.

Example

```
unbind nd6RAvariables -vlan 2 -ipv6Prefix 8000::/64
```

netProfile

Sep 22, 2015

The following operations can be performed on "netProfile":

add | rm | set | unset | show

add netProfile

Creates a net profile. A net profile (or network profile) contains an IP address or an IP set. During communication with physical servers or peers, the NetScaler appliance uses the addresses specified in the profile as the source IP address.

Synopsys

add netProfile <name> [-td <positive_integer>] [-srcIP <string>]

Arguments

name

Name for the net profile. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the profile is created. Choose a name that helps identify the net profile.

td

Traffic Domain Id.

Maximum value: 4094

srcIP

IP address or the name of an IP set.

Example

```
add netProfile prof1 -srcip 10.102.1.10
```

---

rm netProfile

Removes a net profile from the NetScaler appliance.

Synopsys

rm netProfile <name> ...

Arguments

name

Name of the net profile to be removed.

Example

rm netProfile prof1

---

set netProfile

Modifies the srcIP parameter of a net profile.

Synopsys

set netProfile <name> [-srcIP <string>]

Arguments

name

Name of the net profile whose parameter you want to modify.

srcIP

IP address or the name of an IP set.

Example

set netProfile prof_1 -srcIP 10.102.1.10

---

unset netProfile

Removes the srcIP attribute of a net profile..Refer to the set netProfile command for meanings of the arguments.

Synopsys

unset netProfile <name> [-srcIP]

Example

unset netProfile prof1 -srcIP

---

show netProfile

Displays the settings of all net profiles configured on the NetScaler appliance, or of the specified net profile.

Synopsys

show netProfile [<name>]

Arguments

name

Name of the net profile whose details you want to display.

summary

fullValues

format

level

Outputs

srcIP

Source IPaddress or IPSET name.

vpathEncap

enable/disable vPath Encapsulation

td

Traffic Domain Id.

devno

count

stateflag

Example

```
show netProfile
```

netbridge

Sep 22, 2015

The following operations can be performed on "netbridge":

add | rm | show | bind | unbind

add netbridge

Add a network bridge.

Synopsys

add netbridge <name>

Arguments

name

The name of the network bridge.

Example

```
add netbridge bridge1
```

---

rm netbridge

Remove a network bridge.

Synopsys

rm netbridge <name>

Arguments

name

The name of the network bridge.

Example

```
remove netbridge bridge1
```

---

show netbridge

Show configured network bridges.

Synopsys

show netbridge [<name>]

Arguments

name

The name of the network bridge.

format

level

Outputs

tunnel

The name of the tunnel that is a part of this bridge.

vlan

The VLAN that is extended by this network bridge.

IPAddress

The subnet that is extended by this network bridge.

netmask

The network mask for the subnet.

stateflag

Used internally for display.

devno

count

---

bind netbridge

Bind a network bridge to its attributes.

Synopsys

bind netbridge <name> [-tunnel <string> ...] [-vlan <positive_integer> ...] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>]]

Arguments

name

The name of the network bridge.

tunnel

The name of the tunnel that needs to be a part of this network bridge.

vlan

The VLAN that needs to be extended.

Minimum value: 1

Maximum value: 4094

IPAddress

The subnet that needs to be extended.

Example

```
bind netbridge bridge1 -tunnel tun0
```

---

unbind netbridge

Unbind a network bridge from its attributes.

Synopsys

unbind netbridge <name> [-tunnel <string> ...] [-vlan <positive_integer> ...] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>]]

Arguments

name

The name of the network bridge.

tunnel

The name of the tunnel that is part of this network bridge.

vlan

The vlan that is part of this network bridge.

Minimum value: 1

Maximum value: 4094

IPAddress

The subnet that is part of this network bridge.

Example

```
unbind netbridge bridge1 -tunnel tun0
```

---

onLinkIPv6Prefix

Sep 22, 2015

The following operations can be performed on "onLinkIPv6Prefix":

add | rm | set | unset | show

add onLinkIPv6Prefix

add a new on-link global prefix.

Synopsys

add onLinkIPv6Prefix <ipv6Prefix> [-onlinkPrefix (YES | NO)] [-autonomusPrefix (YES | NO)] [-depricatePrefix (YES | NO)] [-decrementPrefixLifeTimes (YES | NO)] [-prefixValideLifeTime <positive_integer>] [-prefixPreferredLifeTime <positive_integer>]

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

onlinkPrefix

RA Prefix onlink flag.

Possible values: YES, NO

Default value: YES

autonomusPrefix

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: YES

depricatePrefix

Depricate the prefix.

Possible values: YES, NO

Default value: NO

decrementPrefixLifeTimes

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: NO

prefixValideLifeTime

Valide life time of the prefix, in seconds.

Default value: 2592000

prefixPreferredLifeTime

Preferred life time of the prefix, in seconds.

Default value: 604800

Example

```
add onLinkIPv6Prefix 8000::/64
```

---

rm onLinkIPv6Prefix

remove an existing on-link global prefix.

Synopsys

```
rm onLinkIPv6Prefix <ipv6Prefix>
```

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

Example

```
rm onLinkIPv6Prefix 8000::/64
```

---

set onLinkIPv6Prefix

set on-link global prefix's configuration variables.

Synopsys

```
set onLinkIPv6Prefix <ipv6Prefix> [-onlinkPrefix ( YES | NO )] [-autonomusPrefix ( YES | NO )] [-depricatePrefix ( YES | NO )] [-decrementPrefixLifeTimes ( YES | NO )] [-prefixValideLifeTime <positive_integer>] [-prefixPreferredLifeTime <positive_integer>]
```

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

onlinkPrefix

RA Prefix onlink flag.

Possible values: YES, NO

Default value: YES

autonomusPrefix

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: YES

depricatePrefix

Depricate the prefix.

Possible values: YES, NO

Default value: NO

decrementPrefixLifeTimes

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: NO

prefixValideLifeTime

Valide life time of the prefix, in seconds.

Default value: 2592000

prefixPreferredLifeTime

Preferred life time of the prefix, in seconds.

Default value: 604800

Example

```
set onLinkIPv6Prefix 8000::/64 -prefixValideLifeTime 2592000
```

---

unset onLinkIPv6Prefix

Use this command to remove onLinkIPv6Prefix settings.Refer to the set onLinkIPv6Prefix command for meanings of the arguments.

Synopsys

```
unset onLinkIPv6Prefix <ipv6Prefix> [-onlinkPrefix] [-autonomusPrefix] [-depricatePrefix] [-decrementPrefixLifeTimes] [-
```


prefixValideLifeTime] [-prefixPreferredLifeTime]

---

show onLinkIPv6Prefix

displays on-link global prefixes.

Synopsys

show onLinkIPv6Prefix [<ipv6Prefix>]

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

format

level

Outputs

onlinkPrefix

RA Prefix onlink flag.

autonomusPrefix

RA Prefix Autonomus flag.

depricatePrefix

Depricate the prefix.

decrementPrefixLifeTimes

RA Prefix Autonomus flag.

prefixValideLifeTime

Valide life time of the prefix, in seconds.

prefixPreferredLifeTime

Preferred life time of the prefix, in seconds.

stateflag

RA Param state flags

prefixCurrValideLfT

Prefix current valid life time

prefixCurrPreferredLfT

Prefix current prefered life time

devno

count

ptp

Sep 22, 2015

The following operations can be performed on "ptp":

set | show

set ptp

Specifies whether to use Precision Time Protocol (PTP) to synchronize time across cluster nodes. This command is applicable in a cluster setup only. If you do not want to use PTP, you must disable PTP, by using this command, and instead enable NTP.

Synopsys

set ptp -state (DISABLE | ENABLE)

Arguments

state

Enables or disables Precision Time Protocol (PTP) on the appliance. If you disable PTP, make sure you enable Network Time Protocol (NTP) on the cluster.

Possible values: DISABLE, ENABLE

Default value: NSA_PTP_ENABLE

---

show ptp

Displays the status of Precision Time Protocol (PTP) on the appliance.

Synopsys

show ptp

Arguments

format

level

Outputs

state

Enables or disables Precision Time Protocol (PTP) on the appliance. If you disable PTP, make sure you enable Network Time Protocol (NTP) on the cluster.

---

rnat

Sep 22, 2015

The following operations can be performed on "rnat":

clear | set | unset | stat | show

clear rnat

Removes an RNAT rule from the NetScaler appliance.

Synopsys

clear rnat ((<network> [<netmask>]) | (<aclname> [-redirectPort])) [-natIP <ip_addr|*>@ ...] [-td <positive_integer>]

Arguments

network

The network address defined for the RNAT entry.

netmask

The subnet mask for the network address.

aclname

An extended ACL defined for the RNAT entry.

redirectPort

The port number to which the packets are redirected.

natIP

The NAT IP address defined for the RNAT entry.

td

Traffic Domain Id.

Maximum value: 4094

---

set rnat

Modifies parameters of an RNAT rule.

Synopsys

set rnat ((<network> [<netmask>] [-natIP <ip_addr|*>@ ...]) | (<aclname> [-redirectPort <port>] [-natIP <ip_addr|*>@ ...])) [-td <positive_integer>]

Arguments

network

IPv4 network address on whose traffic you want the NetScaler appliance to do RNAT processing.

aclname

Name of any configured extended ACL whose action is ALLOW. The condition specified in the extended ACL rule isused as the condition for the RNAT6 rule.

---

unset rnat

Use this command to modify the parameters of configured Reverse NAT on the system..Refer to the set rnat command for meanings of the arguments.

Synopsys

unset rnat ((<network> [<netmask>]) | (<aclname> [-redirectPort])) [-td <positive_integer>] [-natIP <ip_addr|*>@ ...]

---

stat rnat

Display statistics for rnat sessions.

Synopsys

stat rnat [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

Bytes Received (rnatRxBytes)

Bytes received during RNAT sessions.

Bytes Sent (rnatTxBytes)

Bytes sent during RNAT sessions.

Packets Received (rnatRxPkts)

Packets received during RNAT sessions.

Packets Sent (rnatTxPkts)

Packets sent during RNAT sessions.

Syn Sent (rnatTxSyn)

Requests for connections sent during RNAT sessions.

Current RNAT sessions (rnatSessions)

Currently active RNAT sessions.

Example

stat rnat

---

show rnat

Display the Reverse NAT configuration.

Synopsys

show rnat

Arguments

summary

fullValues

format

level

Outputs

network

The network address.

netmask

Subnet mask associated with the network address.

td

Traffic Domain Id.

natIP

Nat IP Address.

aclname

Name of any configured extended ACL whose action is ALLOW. The condition specified in the extended ACL rule isused as the condition for the RNAT6 rule.

redirectPort

The port number to which the packets are redirected.

cfgflags

This contains the flags for RNAT in DB

devno

count

stateflag

rnat6

Sep 22, 2015

The following operations can be performed on "rnat6":

add | bind | unbind | set | unset | clear | show

add rnat6

Adds a Reverse Network Address Translation (RNAT6) rule for IPv6 traffic. When an IPv6 packet generated by a server matches the conditions specified in the RNAT6 rule, the appliance replaces the source IPv6 address of the IPv6 packet with a configured NAT IPv6 address before forwarding it to the destination.

Synopsys

add rnat6 <name> (<network> | (<acl6name> [-redirectPort <port>]))

Arguments

name

Name for the RNAT6 rule. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created. Choose a name that helps identify the RNAT6 rule.

network

IPv6 address of the network on whose traffic you want the NetScaler appliance to do RNAT processing.

acl6name

Name of any configured ACL6 whose action is ALLOW. The rule of the ACL6 is used as an RNAT6 rule.

Example

```
add rnat6 rnat6_name 2002::/64
```

---

bind rnat6

Binds specified IPv6 NAT IPs to an RNAT6 rule.

Synopsys

bind rnat6 <name> <natIP6>@ ...

Arguments

name

Name of the RNAT6 rule to which to bind NAT IPs.

natIP6

One or more IP addresses to be bound to the IP set.

Example

```
bind rnat6 <rnat6_name> <natIP6>@ ...
```

---

unbind rnat6

Unbinds the associated NAT IPv6 address(es) from an RNAT6 rule.

Synopsys

```
unbind rnat6 <name> <natIP6>@ ...
```

Arguments

name

Name of the RNAT6 rule from which to unbind the associated NAT IP address(es).

natIP6

IP address, or multiple addresses, to be unbound from the RNAT6rule. (If using the CLI, use spaces to separate multiple addresses.)

Example

```
unbind rnat6 <rnat6_name> <natIP6>@ ...
```

---

set rnat6

Modifies the specified parameters of an RNAT6 rule.

Synopsys

```
set rnat6 <name> [-redirectPort <port>]
```

Arguments

name

Name of the RNAT6 rule. Required for identifying the RNAT6 rule and cannot be modified.

redirectPort

Port number to which the IPv6 packets are redirected. Applicable to TCP and UDP protocols.

Minimum value: 1

Maximum value: 65535

---

unset rnat6

Resets the specified parameters of an RNAT6 rule to their default settings. Refer to the set rnat6 command for parameter descriptions..Refer to the set rnat6 command for meanings of the arguments.

Synopsys

unset rnat6 <name> [-redirectPort]

---

clear rnat6

Removes an RNAT6 rule from the NetScaler appliance.

Synopsys

clear rnat6 <name>

Arguments

name

Name of the RNAT6 rule to be removed.

---

show rnat6

Displays the settings of all RNAT6 rules configured on the NetScaler appliance, or of the specified RNAT6 rule.

Synopsys

show rnat6 [<name>]

Arguments

name

Name of the RNAT6 rule whose details you want to display.

format

level

Outputs

network

The network address.

acl6name

ACL6 name

natIP6

Nat IP Address.

redirectPort

Redirect Port Value

stateflag

devno

count

rnatip

Sep 22, 2015

The following operations can be performed on "rnatip":

stat rnatip

Display statistics for RNAT sessions.

Synopsys

stat rnatip [<rnatip>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

rnatip

Specifies the NAT IP address of the configured RNAT entry for which you want to see the statistics. If you do not specify an IP address, this displays the statistics for all the configured RNAT entries.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Traffic domain (td)

Traffic domain for ipaddr.

Bytes Received (rxBytes)

Bytes received on this IP address during RNAT sessions.

Bytes Sent (txBytes)

Bytes sent from this IP address during RNAT sessions.

Packets Received (rxPkts)

Packets received on this IP address during RNAT sessions.

Packets Sent (txPkts)

Packets sent from this IP address during RNAT sessions.

Syn Sent (txSyn)

Requests for connections sent from this IP address during RNAT sessions.

Current RNAT sessions (sessions)

Currently active RNAT sessions started from this IP address.

Example

```
stat rnatip 1.1.1.1
```

rnatparam

Sep 22, 2015

The following operations can be performed on "rnatparam":

set | unset | show

set rnatparam

Sets global parameters of RNAT rules on the NetScaler appliance.

Synopsys

set rnatparam -tcpproxy (ENABLED | DISABLED)

Arguments

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

set rnat parameter -tcpproxy ENABLED

---

unset rnatparam

Use this command to remove rnatparam settings.Refer to the set rnatparam command for meanings of the arguments.

Synopsys

unset rnatparam -tcpproxy

---

show rnatparam

Show the rnat parameter.

Synopsys

show rnatparam

Arguments

format

level

Outputs

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Example

```
show rnat parameter
```

route

Sep 22, 2015

The following operations can be performed on "route":

add | clear | rm | set | unset | show

add route

Adds an IPv4 static route to the routing table of the NetScaler appliance.

Synopsys

add route <network> <netmask> <gateway> [-td <positive_integer>] [-distance <positive_integer>] [-cost <positive_integer>] [-weight <positive_integer>] [-advertise (DISABLED | ENABLED)] [-protocol <protocol> ...] [-msr (ENABLED | DISABLED) [-monitor <string>]]

Arguments

network

IPv4 network address for which to add a route entry in the routing table of the NetScaler appliance.

netmask

The subnet mask associated with the network address.

gateway

IP address of the gateway for this route. Can be either the IP address of the gateway, or can be null to specify a null interface route.

cost

The cost of a route is used to compare routes of the same type. The route having the lowest cost is the most preferred route. Possible values: 0 through 65535. Default: 0.

Maximum value: 65535

td

Traffic Domain Id.

Maximum value: 4094

distance

Administrative distance of this route, which determines the preference of this route over other routes, with same destination, from different routing protocols. A lower value is preferred.

Default value: STATIC_ROUTE_DEFAULT_DISTANCE

Maximum value: 255

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: ROUTE_DEFAULT_WEIGHT

Minimum value: 1

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

protocol

Routing protocol used for advertising this route.

Default value: ADV_ROUTE_FLAGS

msr

Monitor this route using a monitor of type ARP or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

add route 10.10.10.0 255.255.255.0 10.10.10.1

---

clear route

Removes routes of the specifiedtype(protocol) from the routing table of the NetScaler appliance.

Synopsys

clear route <routeType>

Arguments

routeType

Protocol used by routes that you want to remove from the routing table of the NetScaler appliance.

---

rm route

Removes a static route from the NetScaler appliance. Note: You cannot use this command to remove routes that are part of a VLAN configuration. Use the rmvlan or clear vlan command instead.

Synopsys

rm route <network> <netmask> <gateway> [-td <positive_integer>]

Arguments

network

Network address specified in the route entry that you want to remove from the routing table of the NetScaler appliance.

netmask

Subnet mask associated with the network address.

gateway

IP address of the gateway for this route.

td

The Traffic Domain Id of the route to be removed.

Maximum value: 4094

---

set route

Modifies parameters of an IPv4 static route.

Synopsys

set route <network> <netmask> <gateway> [-td <positive_integer>] [-distance <positive_integer>] [-cost <positive_integer>] [-weight <positive_integer>] [-advertise (DISABLED | ENABLED)] [-protocol <protocol> ...] [-msr (ENABLED | DISABLED) [-monitor <string>]]

Arguments

network

Network address in the route entry that you want to modify.

netmask

Subnet mask associated with the network address.

gateway

IP address of the gateway for this route. Can be either the IP address of the gateway, or can be null to specify a null interface route.

td

Traffic Domain Id.

Maximum value: 4094

distance

Administrative distance of this route, which determines the preference of this route over other routes, with same destination, from different routing protocols. A lower value is preferred.

Default value: STATIC_ROUTE_DEFAULT_DISTANCE

Maximum value: 255

cost

The cost of a route is used to compare routes of the same type. The route having the lowest cost is the most preferred route. Possible values: 0 through 65535. Default: 0.

Maximum value: 65535

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: ROUTE_DEFAULT_WEIGHT

Minimum value: 1

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

protocol

Routing protocol used for advertising this route.

Default value: ADV_ROUTE_FLAGS

msr

Monitor this route using a monitor of type ARP or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set route 10.10.10.0 255.255.255.0 10.10.10.1 -advertise enable
```

---

unset route

Unset the attributes of a route that were added by the add/set route command..Refer to the set route command for meanings of the arguments.

Synopsys

```
unset route <network> <netmask> <gateway> [-td <positive_integer>] [-advertise] [-distance] [-cost] [-weight] [-protocol] [-msr] [-monitor]
```

Example

```
unset route 10.10.10.0 255.255.255.0 10.10.10.1 -advertise enable
```

---

show route

Display the configured routing information.

Synopsys

```
show route [<network> <netmask> [<gateway>] [-td <positive_integer>]] [<routeType>] [-detail]
```

Arguments

network

The destination network or host.

routeType

The type of routes to be shown.

detail

Display a detailed view.

summary

fullValues

format

level

Outputs

gatewayName

The name of the gateway for this route. For a route other than a link load balancing (LLB) route, this value is null.

advertise

Enable advertisement.

type

State of the RNAT.

stateflag

dynamic

State of the route.

STATIC

PERMANENT

DIRECT

NAT

LBROUTE

ADV

TUNNEL

Show whether it is a tunnel route or not.

cost

The cost of a route is used to compare routes of the same type. The route having the lowest cost is the most preferred route. Possible values: 0 through 65535. Default: 0.

distance

Administrative distance of this route, which determines the preference of this route over other routes, with same destination, from different routing protocols. A lower value is preferred.

weight

The weight of this route.

protocol

Routing protocol used for advertising this route.

data

Internal data of this route.

data0

Internal route type is stored, used for get.

flags

If this route is dynamic, the name of the routing protocol from which it was learned.

routeOwners

Use this option with -dynamic and in a cluster only to specify the set of nodes from which this dynamic route has been learnt.

retain

OSPF

OSPF protocol.

ISIS

ISIS protocol.

RIP

RIP protocol.

BGP

BGP protocol.

DHCP

advOSPF

Advertised through OSPF protocol.NOTE: This attribute is deprecated.This argument is deprecated.Use protocol parameter to read advertise properties of route

advISIS

Advertised through ISIS protocol.NOTE: This attribute is deprecated.This argument is deprecated.Use protocol parameter to read advertise properties of route

advRIP

Advertised through RIP protocol.NOTE: This attribute is deprecated.This argument is

deprecated.Use protocol parameter to read advertise properties of route

advBGP

Advertised through BGP protocol.NOTE: This attribute is deprecated.This argument is deprecated.Use protocol parameter to read advertise properties of route

msr

Whether MSR is enabled or disabled.

monitor

Name of the monitor, of type ARP or PING, configured on the NetScaler appliance to monitor this route.

state

The state of the static route. Possible values: UP, DOWN.

peFlags

PE flags.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter used with the message code.

monStatParam2

Second parameter used with the message code.

monStatParam3

Third parameter used with the message code.

devno

count

Example

An example of the output of the show route command is as follows: 3 config

route6

Sep 22, 2015

The following operations can be performed on "route6":

add | clear | rm | set | unset | show

add route6

Adds an IPv6 static route to the routing table of the NetScaler appliance.

Synopsys

add route6 <network> [<gateway>] [-vlan <positive_integer>] [-weight <positive_integer>] [-distance <positive_integer>] [-cost <positive_integer>] [-advertise (DISABLED | ENABLED)] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-td <positive_integer>]

Arguments

network

IPv6 network address for which to add a route entry to the routing table of the NetScaler appliance.

gateway

The gateway for this route. The value for this parameter is either an IPv6 address or null.

vlan

Integer value that uniquely identifies a VLAN through which the NetScaler appliance forwards the packets for this route.

Maximum value: 4094

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: 1

Minimum value: 1

Maximum value: 65535

distance

Administrative distance of this route from the appliance.

Default value: 1

Minimum value: 1

Maximum value: 254

cost

Positive integer used by the routing algorithms to determine preference for this route. The lower the cost, the higher the preference.

Default value: 1

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

msr

Monitor this route witha monitor of type ND6 or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

td

Traffic Domain Id for IPv6 network.

Maximum value: 4094

Example

```
add route6 ::/0 2004::1  add route6 ::/0 FE80::67 -vlan 5
```

---

clear route6

Removes IPv6 routes of the specified type (protocol) from the routing table of the NetScaler appliance.

Synopsys

clear route6 <routeType>

Arguments

routeType

Type of IPv6 routes to remove from the routing table of the NetScaler appliance.

---

rm route6

Removes a static IPv6 route from the NetScaler appliance.

Synopsys

rm route6 <network> [<gateway>] [-vlan <positive_integer>] [-td <positive_integer>]

Arguments

network

The network of the route to be removed.

gateway

The gateway address of the route to be removed.

vlan

Integer that uniquely identifies the VLAN defined for this route.

Maximum value: 4094

td

Traffic Domain Id for IPv6 network.

Maximum value: 4094

Example

rm route6 ::/0 2004::1 rm route6 ::/0 FE80::67 -vlan 5

---

set route6

Modifies parameters of an IPv6 static route.

Synopsys

set route6 <network> [<gateway>] [-vlan <positive_integer>] [-weight <positive_integer>] [-distance <positive_integer>] [-cost <positive_integer>] [-advertise (DISABLED | ENABLED)] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-td <positive_integer>]

Arguments

network

IPv6 network address of the route entry to be modified.

gateway

The gateway for the route's destination network.

vlan

Integer value that uniquely identifies a VLAN through which the NetScaler appliance forwards the packets for this route.

Maximum value: 4094

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: 1

Minimum value: 1

Maximum value: 65535

distance

Administrative distance of this route from the appliance.

Default value: 1

Minimum value: 1

Maximum value: 254

cost

Positive integer used by the routing algorithms to determine preference for this route. The lower the cost, the higher the preference.

Default value: 1

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

msr

Monitor this route witha monitor of type ND6 or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

td

Traffic Domain Id for IPv6 network.

Maximum value: 4094

Example

```
set route6 1::1/100 2000::1 -advertise enable
```

---

unset route6

Unset the attributes of a route that were added by the add/set route command..Refer to the set route6 command for meanings of the arguments.

Synopsys

```
unset route6 <network> [<gateway>] [-vlan <positive_integer>] [-td <positive_integer>] [-weight] [-distance] [-cost] [-advertise] [-msr] [-monitor]
```

Example

```
unset route6 2000::1/100 3000::1 -advertise enable
```

---

show route6

Displays configuration and state information of all IPv6 routes in the NetScaler appliance's routing table, or of the specified IPv6 route.

Synopsys

```
show route6 [<network> [<gateway>] [-vlan <positive_integer>] [-td <positive_integer>]] [<routeType>] [-detail]
```

Arguments

network

IPv6 network address of the route entry for which to display details.

routeType

The type of IPv6 routes to be to be displayed.

detail

To get a detailed view.

summary

fullValues

format

level

Outputs

gatewayName

The name of the gateway for this route.

advertise

Any gateway of the route entry for which the details are to be displayed.

type

State of the RNAT.

stateflag

dynamic

Whether this route is dynamically learned or not.

weight

Weight of this route.

distance

Administrative distance of this route from the appliance.

cost

Positive integer used by the routing algorithms to determine preference for this route. The lower the cost, the higher the preference.

data

Internal data of this route.NOTE: This attribute is deprecated.This option is deprecated in favour of NSA_DATA1

flags

For a dynamic route, the routing protocol from which the route was learned.

msr

Whether MSR is enabled or disabled.

monitor

Name of the monitor, of type ND6 or PING, configured on the NetScaler appliance to monitor this route.

state

Whether this route is UP or DOWN.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Current number of failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

data1

Internal data of this route.

routeOwners

Use this option with -dynamic and in a cluster only to specify the set of nodes from which this dynamic route has been learnt.

retain

STATIC

Static route.

PERMANENT

Permanent Route.

connected

Connected Route.

OSPFV3

For a dynamic route, the routing protocol from which the route was learned.

ISIS

If this route is dynamic then which routing protocol was it learnt from.

active

For a dynamic route, the routing protocol from which the route was learned.

BGP

For a dynamic route, the routing protocol from which the route was learned.

RIP

For a dynamic route, the routing protocol from which the route was learned.

raRoute

For a dynamic route, the routing protocol from which the route was learned.

devno

count

Example

Following is an example of the output of the show route6 command: Flags: Static(S), Dynamic(D), Active(A) ----------

rsskeytype

Sep 22, 2015

The following operations can be performed on "rsskeytype":

set | show

set rsskeytype

Synopsys

set rsskeytype -rsstype (ASYMMETRIC | SYMMETRIC)

Arguments

rsstype

Type of RSS key, possible values ASYMMETRIC and SYMMETRIC.

Possible values: ASYMMETRIC, SYMMETRIC

Default value: NSA_RSSKEY_ASYM

---

show rsskeytype

Synopsys

show rsskeytype

Arguments

format

level

Outputs

rsstype

Type of RSS key, possible values ASYMMETRIC and SYMMETRIC.

---

tunnelip

Sep 22, 2015

The following operations can be performed on "tunnelip":

stat tunnelip

Display the statistics related to IP tunnel.

Synopsys

stat tunnelip [<tunnelip>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

tunnelip

remote IP address of the configured tunnel.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received on tunnel (tnlRxPkts)

Total number of packets received on the tunnel.

Packets transmitted on tunnel (tnlTxPkts)

Total number of packets transmitted on the tunnel.

Bytes received on tunnel (tnlRxBytes)

Total number of bytes received on the tunnel.

Bytes transmitted on tunnel (tnlTxBytes)

Total number of bytes transmitted on the tunnel.

Example

```
stat tunnelip 2.1.1.1
```

tunnelip6

Sep 22, 2015

The following operations can be performed on "tunnelip6":

stat tunnelip6

Display the statistics related to IP tunnel.

Synopsys

stat tunnelip6 [<tunnelip6>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

tunnelip6

remote IPv6 address of the configured tunnel.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received on tunnel (tnlRxPkts)

Total number of packets received on the tunnel.

Packets transmitted on tunnel (tnlTxPkts)

Total number of packets transmitted on the tunnel.

Bytes received on tunnel (tnlRxBytes)

Total number of bytes received on the tunnel.

Bytes transmitted on tunnel (tnlTxBytes)

Total number of bytes transmitted on the tunnel.

Example

```
stat tunnelip6 2001::1
```

vPathParam

Sep 22, 2015

The following operations can be performed on "vPathParam":

set | unset | show

set vPathParam

Sets the global parameters for vPath

Synopsys

set vPathParam [-srcIP <ip_addr>] [-offload (ENABLED | DISABLED)]

Arguments

srcIP

source-IP address used for all vPath L3 encapsulations. Must be a MIP or SNIP address.

offload

enable/disable vPath offload feature

Possible values: ENABLED, DISABLED

Default value: 2

Example

```
set vpathparam -srcip 2.2.2.2
```

---

unset vPathParam

Use this command to remove vPathParam settings.Refer to the set vPathParam command for meanings of the arguments.

Synopsys

unset vPathParam [-srcIP] [-offload]

---

show vPathParam

Display the global parameters for vPath

Synopsys

show vPathParam

Arguments

format

level

Outputs

srcIP

srcIP used for vPath encapsulation.

Encapsulation

Global vPath encapsulation .

offload

enable/disable vPath offload feature

Example

show vpathparam

---

vlan

Sep 22, 2015

The following operations can be performed on "vlan":

add | rm | set | unset | bind | unbind | show | stat

add vlan

Adds a VLAN to the NetScaler appliance.The newVLAN is not active unless interfaces are bound to it.

Synopsys

add vlan <id> [-aliasName <string>] [-ipv6DynamicRouting (ENABLED | DISABLED)]

Arguments

id

ID of the VLAN whose parameters you want to modify.

Minimum value: 1

Maximum value: 4094

aliasName

A name for the VLAN. Must begin with a letter, a number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the VLAN. However, you cannot perform any VLAN operation by specifying this name instead of the VLAN ID.

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this VLAN. Note: For the ENABLED setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line.

Possible values: ENABLED, DISABLED

Default value: DISABLED

---

rm vlan

Removes a VLAN from the NetScaler appliance. When the VLAN is removed, its interfaces are bound to VLAN 1. Note: VLAN 1 cannot be removed by any command.

Synopsys

rm vlan <id>

Arguments

id

Integer that uniquely identifies the VLAN to be removed from the NetScaler appliance. When the VLAN is removed, its interfaces become members of VLAN 1.

Minimum value: 2

Maximum value: 4094

---

set vlan

Modifies parameters of a VLAN on the NetScaler appliance.

Synopsys

set vlan <id> [-aliasName <string>] [-ipv6DynamicRouting (ENABLED | DISABLED)]

Arguments

id

ID of the VLAN whose parameters you want to modify.

Minimum value: 1

Maximum value: 4094

aliasName

A name for the VLAN. Must begin with a letter, a number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the VLAN. However, you cannot perform any VLAN operation by specifying this name instead of the VLAN ID.

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this bridge group. Note: For the ENABLED setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

set vlan 2 -dynamicRouting ENABLED

---

unset vlan

Use this command to remove vlan settings.Refer to the set vlan command for meanings of the arguments.

Synopsys

unset vlan <id> [-aliasName] [-ipv6DynamicRouting]

---

bind vlan

Binds the specified interfaces or IP addresses to a VLAN. An interface can be bound to a VLAN as a tagged or an untagged member. Adding an interface as an untagged member removes it from its current native VLAN and adds it to the new VLAN. If an interface is added as a tagged member to a VLAN, it still remains a member of its native VLAN.

Synopsys

bind vlan <id> [-ifnum <interface_name> ... [-tagged]] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]

Arguments

id

Specifies the virtual LAN ID.

Minimum value: 1

Maximum value: 4094

ifnum

Interface to be bound to the VLAN, specified in slot/port notation (for example, 1/3).

Minimum value: 1

IPAddress

Network address to be associated with the VLAN. Should exist on the appliance before you associate it with the VLAN. To enable IP forwarding among VLANs, the specified address can be used as the default gateway by the hosts in the network.

---

unbind vlan

Unbinds the specified interfaces or IP addresses from a VLAN. If any of the interfaces are untagged members of the VLAN, they are automatically bound to VLAN 1.

Synopsys

unbind vlan <id> [-ifnum <interface_name> ... [-tagged]] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]

Arguments

id

The virtual LAN (VLAN) id.

Minimum value: 1

Maximum value: 4094

ifnum

Interface to unbind from the VLAN, specified in slot/port notation (for example, 1/3).

Minimum value: 1

IPAddress

The IP Address associated with the VLAN configuration.

---

show vlan

Displays the settings of all VLANs configured on the NetScaler appliance, or of the specified VLAN. To display the settings of all the VLANs, run the command without any parameters. To display the settings of a particular VLAN, specify the ID of the VLAN.

Synopsys

show vlan [<id>] show vlan stats - alias for 'stat vlan'

Arguments

id

Integer that uniquely identifies the VLAN for which the details are to be displayed.

Minimum value: 1

Maximum value: 4094

summary

fullValues

format

level

Outputs

aliasName

A name for the VLAN. Must begin with a letter, a number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the VLAN. However, you cannot perform any VLAN operation by specifying this name instead of the VLAN ID.

IPAddress

The IP address assigned to the VLAN.

netmask

Subnet mask for the network address defined for this VLAN.

linklocalIPv6Addr

The link-local IP address assigned to the VLAN.

rnat

Temporary flag used for internal purpose.

stateflag

state flag

portbitmap

Member interfaces of this vlan.

lsbitmap

Member linksets of this vlan.

tagbitmap

Tagged members of this vlan.

lstagbitmap

Tagged linksets of this vlan.

ifaces

Names of all member interfaces of this vlan.

taglfaces

Names of all tagged member interfaces of this vlan.

ipv6DynamicRouting

Whether dynamic routing is enabled or disabled.

flag

ifnum

The interface to be bound to the VLAN, specified in slot/port notation (for example, 1/3).

tagged

Make the interface an 802.1q tagged interface. Packets sent on this interface on this VLAN have an additional 4-byte 802.1q tag, which identifies the VLAN. To use 802.1q tagging, you must also configure the switch connected to the appliance's interfaces.

td

Traffic Domain Id.

sdxVlan

SDX vlan.

devno

count

Example

An example of the output of the show vlan command is as follows: 1) VLAN ID: 5 VLAN Alias Name: Interfaces : 1/7 IPs

---

stat vlan

Display statistics for VLAN(s).

Synopsys

stat vlan [<id>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

id

An integer specifying the VLAN identification number (VID). Possible values: 1 through 4094.

Minimum value: 1

Maximum value: 4094

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received (RxPkts)

Packets received on the VLAN.

Bytes received (RxBytes)

Bytes of data received on the VLAN.

Packets sent (TxPkts)

Packets transmitted on the VLAN.

Bytes sent (TxBytes)

Bytes of data transmitted on the VLAN.

Packets dropped (DropPkts)

Inbound packets dropped by the VLAN upon reception.

received (BcastPkt)

Broadcast packets sent and received on the VLAN.

Example

stat vlan 1

---

vpath

Sep 22, 2015

The following operations can be performed on "vpath":

add | rm | show | stat

add vpath

Adds vPath destination IP to which packets need to be vPath injected.

Synopsys

add vpath <name> (<destIP> [<netmask>] [<gateway>])

Arguments

name

Name for the vPath. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the profile is created. Choose a name that helps identify the net profile.

destIP

This is the destination ip, where vPath encapsulated packets needs to be sent

Example

```
add vpath vPath1 -destip 10.102.1.10
```

---

rm vpath

Remove vPath destination IP.

Synopsys

rm vpath <name> ...

Arguments

name

Name of the vPath to be removed.

Example

```
rm netProfile prof1
```

---

show vpath

List down all vPath destination IPs.

Synopsys

show vpath [<name>]

Arguments

name

Name of the vPath whose details you want to display.

summary

fullValues

format

level

Outputs

destIP

This is the destination ip, where vPath encapsulated packets needs to be sent

netmask

Subnet mask associated with the destination network.

gateway

Next hop gateway to reach the destination address.

devno

count

stateflag

Example

show vpath

---

stat vpath

Display vPath statistics.

Synopsys

stat vpath [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

L2 data packets received (TotL2DataRx)

Total number of non-fragmented vPath data packets decapsulated in L2 adjacency

L3 data packets received (TotL3DataRx)

Total number of non-fragmented vPath data packets decapsulated in L3 adjacency

L2 control packets received (TotL2CntrlPkts)

Total number of vPath control packets received in L2 adjacency

L3 control packets received (TotL3CntrlPkts)

Total number of vPath control packets received in L3 adjacency

Fragmented packets received (TotFragPkts)

Total number of vPath fragments received

L2 packets transmitted (TotL2EncapPkts)

Total number of L2 vPath encapsulated packets injected to VEM

L3 packets transmitted (TotL3EncapPkts)

Total number of L3 vPath encapsulated packets injected to VEM

Fragmented packets transmitted (TotFragEncapPkts)

Number of fragmented vPath packets transmitted

Offload packets transmitted (TotOffload)

Number of offloaded vPath packets transmitted

vrID

Sep 22, 2015

The following operations can be performed on "vrID":

add | rm | set | unset | bind | unbind | show

Adds a VMAC address to the NetScaler appliance. A Virtual MAC address (VMAC) is a floating entity, shared by the nodes in an HA configuration.

add vrID <id> [-priority <positive_integer>] [-preemption (ENABLED | DISABLED)] [-sharing (ENABLED | DISABLED)] [-tracking <tracking>]

id

Integer that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

priority

Base priority (BP), in an active-active mode configuration, which ordinarily determines the master VIP address.

Default value: 255

Minimum value: 1

Maximum value: 255

preemption

In an active-active mode configuration, make a backup VIP address the master if its priority becomes higher than that of a master VIP address bound to this VMAC address.

If you disable pre-emption while a backup VIP address is the master, the backup VIP address remains master until the original master VIP's priority becomes higher than that of the current master.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sharing

In an active-active mode configuration, enable the backup VIP address to process any traffic instead of dropping it.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tracking

The effective priority (EP) value, relative to the base priority (BP) value in an active-active mode configuration. When EP is set to a value other than None, it is EP, not BP, which determines the master VIP address.

Available settings function as follows:

* NONE - No tracking. EP = BP

* ALL - If the status of all virtual servers is UP, EP = BP. Otherwise, EP = 0.

* ONE - If the status of at least one virtual server is UP, EP = BP. Otherwise, EP = 0.

* PROGRESSIVE - If the status of all virtual servers is UP, EP = BP. If the status of all virtual servers is DOWN, EP = 0. Otherwise EP = BP (1 - K/N), where N is the total number of virtual servers associated with the VIP address and K is the number of virtual servers for which the status is DOWN.

Default: NONE.

Possible values: NONE, ONE, ALL, PROGRESSIVE

Default value: TRACK_NONE

add vrID 1

Removes a specified VMAC entry or all VMAC entries from the NetScaler appliance.

rm vrID (<id> | -all)

id

Integer value that uniquely identifies the VMAC address.

Minimum value: 1

Maximum value: 255

all

Remove all the configured VMAC addresses from the NetScaler appliance.

Modifies parameters related to a VMAC address on the NetScaler appliance.

set vrID <id> [-priority <positive_integer>] [-preemption (ENABLED | DISABLED)] [-sharing (ENABLED | DISABLED)] [-tracking <tracking>]

id

Integer value that uniquely identifies the VMAC address. The generic VMACaddressis in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

priority

Base priority (BP), in an active-active mode configuration, which ordinarily determines the master VIP address.

Default value: 255

Minimum value: 1

Maximum value: 255

preemption

In an active-active mode configuration, make a backup VIP address the master if its priority becomes higher than that of a master VIP address bound to this VMAC address.

If you disable pre-emption while a backup VIP address is the master, the backup VIP address remains master until the original master VIP's priority becomes higher than that of the current master.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sharing

In an active-active mode configuration, enable the backup VIP address to process any traffic instead of dropping it.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tracking

The effective priority (EP) value, relative to the base priority (BP) value in an active-active mode configuration. When

EP is set to a value other than None, it is EP, not BP, which determines the master VIP address.

Available settings function as follows:

* NONE - No tracking. EP = BP

* ALL - If the status of all virtual servers is UP, EP = BP. Otherwise, EP = 0.

* ONE - If the status of at least one virtual server is UP, EP = BP. Otherwise, EP = 0.

* PROGRESSIVE - If the status of all virtual servers is UP, EP = BP. If the status of all virtual servers is DOWN, EP = 0. Otherwise EP = BP (1 - K/N), where N is the total number of virtual servers associated with the VIP address and K is the number of virtual servers for which the status is DOWN.

Default: NONE.

Possible values: NONE, ONE, ALL, PROGRESSIVE

Default value: TRACK_NONE

```
set vrID 1 -priority 100
```

Use this command to remove vrID settings.Refer to the set vrID command for meanings of the arguments.

```
unset vrID <id> [-priority] [-preemption] [-sharing] [-tracking]
```

Binds the specified interfaces to a VMAC configuration.

```
bind vrID <id> -ifnum <interface_name> ...
```

id

Integer that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to bind to the VMAC, specified in (slot/port) notation (for example, 1/2).Use spaces to separate multiple entries.

add vrID 1

Unbinds specified interfaces from a VMAC configuration.

unbind vrID <id> -ifnum <interface_name> ...

id

Integer value that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to unbind from the VMAC, specified in (slot/port) notation (for example, 1/2). Use spaces to separate multiple entries.

Displays the settings of all VRIDs configured on the NetScaler appliance, or of the specified VRID. To display the settings of all the VRIDs, run the command without any parameters. To display the settings of a particular VRID, specify the VRID.

show vrID [<id>]

id

Integer value that uniquely identifies the VMAC address.

Minimum value: 1

Maximum value: 255

summary

fullValues

format

level

ifaces

Interfaces bound to this VRID.

type

Indicates whether this VRID entry was added manually or dynamically. When you manually add a VRID entry, the value for this parameter is STATIC. Otherwise, it is DYNAMIC.

vlan

The VLAN in which this VRID resides.

priority

Base priority (BP), in an active-active mode configuration, which ordinarily determines the master VIP address.

effectivePriority

The effective priority of this VRID.

preemption

In an active-active mode configuration, make a backup VIP address the master if its priority becomes higher than that of a master VIP address bound to this VMAC address.

If you disable pre-emption while a backup VIP address is the master, the backup VIP address remains master until the original master VIP's priority becomes higher than that of the current master.

sharing

In an active-active mode configuration, enable the backup VIP address to process any traffic instead of dropping it.

tracking

The effective priority (EP) value, relative to the base priority (BP) value in an active-active mode configuration. When EP is set to a value other than None, it is EP, not BP, which determines the master VIP address.

Available settings function as follows:

* NONE - No tracking. EP = BP

* ALL - If the status of all virtual servers is UP, EP = BP. Otherwise, EP = 0.

* ONE - If the status of at least one virtual server is UP, EP = BP. Otherwise, EP = 0.

* PROGRESSIVE - If the status of all virtual servers is UP, EP = BP. If the status of all virtual servers is DOWN, EP = 0. Otherwise EP = BP (1 - K/N), where N is the total number of virtual servers associated with the VIP address and K is the number of virtual servers for which the status is DOWN.

Default: NONE.

flags

Flags.

IPAddress

The IP address bound to the VRID.

state

State of this VRID.

stateflag

devno

count

show vrid

vrID6

Sep 22, 2015

The following operations can be performed on "vrID6":

add | rm | bind | unbind | show

Adds a VMAC6 address to the NetScaler appliance. A Virtual MAC address (VMAC6) is a floating entity, shared by the nodes in an HA configuration.

add vrID6 <id>

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

add vrID6 1

Removes a specified VMAC6 entry or all VMAC6 entries from the NetScaler appliance.

rm vrID6 (<id> | -all)

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

all

Remove all configured VMAC6 addresses from the NetScaler appliance.

Binds the specified interfaces to a VMAC6 configuration.

bind vrID6 <id> -ifnum <interface_name> ...

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to bind tothe VMAC6, specified in (slot/port) notation (for example, 1/2).Use spaces to separate multiple entries.

add vrID6 1

Unbinds the specified interfaces from a VMAC6 configuration.

unbind vrID6 <id> -ifnum <interface_name> ...

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to unbind from the VMAC6, specified in (slot/port) notation (for example, 1/2). Use spaces to separate multiple entries.

Displays the settings of all VRID6s configured on the NetScaler appliance, or of the specified VRID6. To display the settings of all the VRID6s, run the command without any parameters. To display the settings of a particular VRID6, specify the VRID6.

show vrID6 [<id>]

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

summary

fullValues

format

level

ifaces

Interfaces bound to this VRID.NOTE: This attribute is deprecated.This argument is deprecated and is replaced by the -ifnum argument.

ifnum

Interfaces bound to this vrid.

type

Type (static or dynamic) of this VRID.

vlan

The VLAN in which this VRID resides.

priority

The priority of this VRID.

state

State of this VRID.

flags

Flags.

stateflag

IPAddress

The IP address bound to the VRID6

devno

count

show vrid6

vrIDParam

Sep 22, 2015

The following operations can be performed on "vrIDParam":

set | unset | show

Sets global parameters of VMACs on the NetScaler appliance.

set vrIDParam -sendToMaster (ENABLED | DISABLED)

sendToMaster

Forward packets to the master node, in an active-active mode configuration, if the virtual server is in the backup state and sharing is disabled.

Possible values: ENABLED, DISABLED

Default value: DISABLED

set vrIDParam -sendToMaster ENABLED

Use this command to remove vrIDParam settings.Refer to the set vrIDParam command for meanings of the arguments.

unset vrIDParam -sendToMaster

Displays the VRID global settings on the NetScaler appliance.

show vrIDParam

format

level

sendToMaster

Forward packets to the master node, in an active-active mode configuration, if the virtual server is in the backup state and sharing is disabled.

NS Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- ns
- ns acl
- ns acl6
- ns acls
- ns acls6
- ns appflowCollector
- ns appflowParam
- ns aptlicense
- ns config
- ns connectiontable
- ns consoleloginprompt
- ns dhcpIp
- ns dhcpParams
- ns diameter
- ns encryptionParams
- ns events
- ns feature
- ns hardware
- ns hostName
- ns httpParam
- ns httpProfile
- ns idletimeout
- ns info
- ns ip
- ns ip6
- ns license
- ns limitIdentifier
- ns limitSelector
- ns limitSessions
- ns memory
- ns mode
- ns ns.conf
- ns param
- ns pbr
- ns pbr6
- ns pbrs
- ns persistencesession
- ns rateControl
- ns rollbackcmd
- ns rpcNode
- ns runningConfig

- ns savedConfig
- ns simpleacl
- ns simpleacl6
- ns spParams
- ns stats
- ns surgeQ
- ns tcpParam
- ns tcpProfile
- ns tcpbufParam
- ns timeout
- ns timer
- ns trafficDomain
- ns version
- ns weblogparam
- ns xmlnamespace
- reboot
- shutdown

ns

Sep 22, 2015

The following operations can be performed on "ns":

config | stat

Displays a menu to configure the basic parameters of a NetScaler appliance. Note: The appliance must be rebooted for these changes to take effect.

config ns

Displays generic statistics of the NetScaler appliance.

stat ns [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

Maximum memory(KB) Deprecated (MaxMem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Delta compression ratio (DlCmpRt)

Ratio of compressible data received to compressed data transmitted.If this ratio is one (uncmp:1.0) that means compression is disabled or we are not able to compress even a single compressible packet.

Average CPU usage (CPU)

Shows average CPU utilization percentage if more than 1 CPU is present.

CPU usage (CPU)

CPU utilization: percentage * 10.

Memory usage (MemUsage)

Percentage of memory utilization on NetScaler.

Total HTTP compression ratio

Ratio of total HTTP data received to total HTTP data transmitted.

HTTP compression ratio

Ratio of the compressible data received from the server to the compressed data sent to the client.

Utilized memory(KB) (UtiMem)

Amount of memory the integrated cache is currently using.

Maximum memory active value(KB) (MaxMemActive)

Currently active value of maximum memory

Maximum memory(KB) (Max64Mem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Origin bandwidth saved(%) (POrBan)

Percentage of origin bandwidth saved, expressed as number of bytes served from the integrated cache divided by all bytes served. The assumption is that all compression is done in the NetScaler.

Misses (TotMiss)

Intercepted HTTP requests requiring fetches from origin server.

Hits (TotHit)

Responses served from the integrated cache. These responses match a policy with a CACHE action.

SSL cards UP (SSLCardUP)

Number of SSL cards that are UP. If the number of cards UP is lower than a threshold, a failover is initiated.

InUse Memory (%) (MemUsage)

Percentage of memory utilization on NetScaler.

Memory usage (MB) (MemUseMB)

Main memory currently in use, in megabytes.

Management CPU usage (%) (CPU)

Management CPU utilization percentage.

Packet CPU usage (%) (CPU)

Average CPU utilization percentage for all packet engines excluding management PE.

Average CPU usage (%) (CPU)

Average CPU utilization percentage. Not applicable for a single-CPU system.

CPU usage (%) (CPU)

CPU utilization percentage.

Up since (Since)

Time when the NetScaler appliance was last started.

Last Transition time (TransTime)

Time when the last master state transition occurred. You can use this statistic for debugging.

System state (HAState)

State of the HA node, based on its health, in a high availability setup. Possible values are:

UP ? Indicates that the node is accessible and can function as either a primary or secondary node.

DISABLED ? Indicates that the high availability status of the node has been manually disabled. Synchronization and propagation cannot take place between the peer nodes.

INIT ? Indicates that the node is in the process of becoming part of the high availability configuration.

PARTIALFAIL ? Indicates that one of the high availability monitored interfaces has failed because of a card or link failure. This state triggers a failover.

COMPLETEFAIL ? Indicates that all the interfaces of the node are unusable, because the interfaces on which high availability monitoring is enabled are not connected or are manually disabled. This state triggers a failover.

DUMB ? Indicates that the node is in listening mode. It does not participate in high availability transitions or transfer configuration from the peer node. This is a configured value, not a statistic.

PARTIALFAILSSL ? Indicates that the SSL card has failed. This state triggers a failover.

ROUTEMONITORFAIL ? Indicates that the route monitor has failed. This state triggers a failover.

Master state (mastate)

Indicates the high availability state of the node. Possible values are:

STAYSECONDARY ? Indicates that the selected node remains the secondary node in a high availability setup. In this case a forced failover does not change the state but, instead, returns an appropriate error message. This is a configured value and not a statistic.

PRIMARY ? Indicates that the selected node is the primary node in a high availability setup.

SECONDARY ? Indicates that the selected node is the secondary node in a high availability setup.

CLAIMING ? Indicates that the secondary node is in the process of taking over as the primary node. This is the intermediate state in the transition of the secondary node to primary status.

FORCE CHANGE - Indicates that the secondary node is forcibly changing its status to primary due to a forced failover issued on the secondary node.

SSL cards present (SSLCards)

Number of SSL crypto cards present on the NetScaler appliance.

/flash Used (%) (disk0PerUsage)

Used space in /flash partition of the disk, as a percentage. This is a critical counter.

You can configure /flash Used (%) by using the Set snmp alarm DISK-USAGE-HIGH command.

/var Used (%) (disk1PerUsage)

Used space in /var partition of the disk, as a percentage. This is a critical counter. You can configure /var Used (%) by using the Set snmp alarm DISK-USAGE-HIGH command.

/flash Available (MB) (disk0Avail)

Available space in /flash partition of the hard disk.

/var Available (MB) (disk1Avail)

Available space in /var partition of the hard disk.

Megabits received (RxMb)

Number of megabytes received by the NetScaler appliance.

Megabits transmitted (TxMb)

Number of megabytes transmitted by the NetScaler appliance.

All client connections (CltCx)

Client connections, including connections in the Opening, Established, and Closing state.

Established client connections (CltCxE)

Current client connections in the Established state, which indicates that data transfer can occur between the NetScaler and the client.

All server connections (SvrCx)

Server connections, including connections in the Opening, Established, and Closing state.

Established server connections (SvrCxE)

Current server connections in the Established state, which indicates that data transfer can occur between the

NetScaler and the server.

Total requests (HTReqRx)

Total number of HTTP requests received.

Total responses (HTRspRx)

Total number of HTTP responses sent.

Request bytes received (HTReqbRx)

Total number of bytes of HTTP request data received.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

SSL transactions (SSLTrn)

Number of SSL transactions on the NetScaler appliance.

SSL session hits (SeHit)

Number of SSL session reuse hits on the NetScaler appliance.

requests (reqs)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

responses (resps)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

aborts

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

redirects (redirect)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

Misc. Counter 0 (misc0)

Miscellaneous Counter 0.

Misc. Counter 1 (misc1)

Miscellaneous Counter 1.

Management CPU usage (CPU)

Management CPU utilization: percentage * 10.

SSL crypto card status (SSLCardSt)

Status of the SSL card(s). The value should be interpreted in binary form, with each set bit indicates a card as UP.

304 hits (304Hit)

Object not modified responses served from the cache. (Status code 304 served instead of the full response.)

Non-304 hits (Non304Hit)

Total number of full (non-304) responses served from the cache. A 304 status code indicates that a response has not been modified since the last time it was served

sql hits (sqlHit)

sql response served from cache

Requests (CacReq)

Total cache hits plus total cache misses.

Compressed bytes transmitted

Number of bytes the NetScaler sends to the client after compressing the response from the server.

Compressible bytes received

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Compressible bytes received (DlCmpRxB)

Total number of delta-compressible bytes received by NetScaler.

Compressed bytes transmitted (DlCmpTxB)

Total number of delta-compressed bytes transmitted by NetScaler.

ns acl

Sep 22, 2015

The following operations can be performed on "ns acl":

add | rm | set | unset | enable | disable | stat | rename | show

Adds an extended ACL rule to the NetScaler appliance. To commit this operation, you must apply the extended ACLs. Extended ACL rules filter data packets on the basis of various parameters, such as IP address, source port, action, and protocol.

add ns acl <aclname> <aclaction> [-td <positive_integer>] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] [-TTL <positive_integer>] [-srcMac <mac_addr>] [(-protocol <protocol> [-established]) | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-icmpType <positive_integer> [-icmpCode <positive_integer>]] [-priority <positive_integer>] [-state (ENABLED | DISABLED)] [-logstate (ENABLED | DISABLED) [-ratelimit <positive_integer>]]

aclname

Name for the extended ACL rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the extended ACL rule is created.

aclaction

Action to perform on incoming IPv4 packets that match the extended ACL rule.

Available settings function as follows:

* ALLOW - The NetScaler appliance processes the packet.

* BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.

* DENY - The NetScaler appliance drops the packet.

Possible values: BRIDGE, DENY, ALLOW

td

Traffic Domain Id.

Maximum value: 4094

srcIP

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destIP

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

TTL

Number of seconds, in multiples of four, after which the extended ACL rule expires. If you do not want the extended ACL rule to expire, do not specify a TTL value.

Minimum value: 1

Maximum value: 2147483647

srcMac

MAC address to match against the source MAC address of an incoming IPv4 packet.

protocol

Protocol to match against the protocol of an incoming IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol to match against the protocol of an incoming IPv4 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance applies the ACL rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL rule to the incoming packets of all interfaces.

established

Allow only incoming TCP packets that have the ACK or RST bit set, if the action set for the ACL rule is ALLOW and these packets match the other conditions in the ACL rule.

icmpType

ICMP Message type to match against the message type of an incoming ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Maximum value: 65536

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

Maximum value: 65536

priority

Priority for the extended ACL rule that determines the order in which it is evaluated relative to the other extended ACL rules. If you do not specify priorities while creating extended ACL rules, the ACL rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 100000

state

Enable or disable the extended ACL rule. After you apply the extended ACL rules, the NetScaler appliance compares incoming packets against the enabled extended ACL rules.

Possible values: ENABLED, DISABLED

Default value: XACLENABLED

logstate

Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.

Possible values: ENABLED, DISABLED

Default value: GENDISABLED

ratelimit

Maximum number of log messages to be generated per second. If you set this parameter, you must enable the Log State parameter.

Default value: 100

Minimum value: 1

Maximum value: 10000

```
add ns acl restrict DENY -srcport 45-1024 -destIP 192.168.1.1 -protocol TCP
```

Removes an extended ACL rule from the NetScaler appliance. To commit this operation, you must apply the extended ACLs.

```
rm ns acl <aclname> ...
```

aclname

Name of the extended ACL rule that you want to remove.

rm ns acl restrict

Modifies the parameters of an ACL rule. To commit this operation, you must apply the extended ACLs.

set ns acl <aclname> [-aclaction <aclaction>] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-icmpType <positive_integer> [-icmpCode <positive_integer>]] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-logstate (ENABLED | DISABLED)] [-ratelimit <positive_integer>] [-established]

aclname

Name of the ACL rule whose parameters you want to modify.

aclaction

Action to perform on incoming IPv4 packets that match the extended ACL rule.

Available settings function as follows:

* ALLOW - The NetScaler appliance processes the packet.

* BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.

* DENY - The NetScaler appliance drops the packet.

Possible values: BRIDGE, DENY, ALLOW

srcIP

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destIP

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcMac

MAC address to match against the source MAC address of an incoming IPv4 packet.

protocol

Protocol to match against the protocol of an incoming IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol to match against the protocol of an incoming IPv4 packet.

Minimum value: 1

Maximum value: 255

icmpType

ICMP Message type to match against the message type of an incoming ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Maximum value: 65536

vlan

ID of the VLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance applies the ACL rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL rule to the incoming packets of all interfaces.

priority

Priority for the extended ACL rule that determines the order in which it is evaluated relative to the other extended ACL rules. If you do not specify priorities while creating extended ACL rules, the ACL rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 100000

logstate

Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.

Possible values: ENABLED, DISABLED

Default value: GENDISABLED

established

Allow only incoming TCP packets that have the ACK or RST bit set, if the action set for the ACL rule is ALLOW and these packets match the other conditions in the ACL rule.

```
set ns acl restrict -srcPort 50
```

Resets the attributes of the specified extended ACL rule. Attributes for which a default value is available revert to their default values. Refer to the set ns acl command for a description of the parameters..Refer to the set ns acl command for meanings of the arguments.

```
unset ns acl <aclname> [-srcIP] [-srcPort] [-destIP] [-destPort] [-srcMac] [-protocol] [-icmpType] [-icmpCode] [-vlan] [-interface] [-logstate] [-ratelimit] [-established]
```

```
unset ns acl rule1 -srcPort
```

Enables an extended ACL rule. To commit this operation, you must apply the extended ACLs. After you apply the extended ACL rules, the NetScaler appliance compares incoming packets against the enabled extended ACL rules.

```
enable ns acl <aclname> ...
```

aclname

Name of the extended ACL rule that you want to enable.

```
enable ns acl foo
```

Disables an extended ACL rule. To commit this operation, you must apply the extended ACLs. After you apply the ACL rules, the NetScaler appliance does not compare incoming packets against the disabled extended ACL rules.

```
disable ns acl <aclname> ...
```

aclname

Name of the extended ACL rule that you want to disable.

disable ns acl foo

Displays statistics related to the extended ACL rules. To display statistics of all the extended ACL rules, run the command without any parameters. To display statistics of a particular extended ACL rule, specify the name of the extended ACL rule.

stat ns acl [<aclname>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

aclname

Name of the extended ACL rule whose statistics you want the NetScaler appliance to display.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Bridge ACL hits (ACLBdg)

Packets matching a bridge ACL, which is in transparent mode and bypasses service processing.

Deny ACL hits (ACLDeny)

Packets dropped because they match ACLs with processing mode set to DENY.

Allow ACL hits (ACLAllow)

Packets matching ACLs with processing mode set to ALLOW. NetScaler processes these packets.

NAT ACL hits (ACLNAT)

Packets matching a NAT ACL, resulting in a NAT session.

ACL hits (ACLHits)

Packets matching an ACL.

ACL misses (ACLMiss)

Packets not matching any ACL.

ACL Count (ACLCount)

Total number of ACL rules configured.

Hits for this ACL (Hits)

Number of times the acl was hit

stat acl

Renames an extended ACL rule.

rename ns acl <aclname> <newName>

aclname

Name of the extended ACL rule that you want to rename.

newName

New name for the extended ACL rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

rename acl rule rule-new

Displays settings related to the extended ACL rules. To display settings of all the extended ACL rules, run the command without any parameters. To display settings of a particular extended ACL rule, specify the name of the extended ACL rule.

show ns acl [<aclname>]

aclname

Name of the extended ACL rule whose details you want the NetScaler appliance to display.

summary

fullValues

format

level

td

Traffic Domain Id.

aclaction

Action to perform on incoming IPv4 packets that match the extended ACL rule.

Available settings function as follows:

* ALLOW - The NetScaler appliance processes the packet.

* BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.

* DENY - The NetScaler appliance drops the packet.

srcMac

MAC address to match against the source MAC address of an incoming IPv4 packet.

stateflag

ACL state flag.

protocol

The protocol number in IP header or name.

protocolNumber

The protocol number in IP header or name.

srcPortVal

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destPortVal

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcIPVal

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

vlan

ID of the VLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL rule to the incoming packets on all VLANs.

state

Enable or disable the extended ACL rule. After you apply the extended ACL rules, the NetScaler appliance compares incoming packets against the enabled extended ACL rules.

TTL

Number of seconds, in multiples of four, after which the extended ACL rule expires. If you do not want the extended ACL rule to expire, do not specify a TTL value.

icmpType

ICMP Message type to match against the message type of an incoming ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

interface

ID of an interface. The NetScaler appliance applies the ACL rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL rule to the incoming packets of all interfaces.

hits

The hits of this ACL.

established

This flag indicates that the ACL should be used for TCP response traffic only.

priority

Priority for the extended ACL rule that determines the order in which it is evaluated relative to the other extended ACL rules. If you do not specify priorities while creating extended ACL rules, the ACL rules are evaluated in the order in which they are created.

operator

Either the equals (=) or does not equal (!=) logical operator.

kernelstate

The commit status of the ACL.

logstate

Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.

ratelimit

Packet rate limit for acl logging

time

Time when this acl is applied.

devno

count

```
sh acl foo  TD: 100    Name: foo                  Action: ALLOW   Hits: 0    srcIP = 10.102.1.150    destIP = 202.54.12.47
```

ns acl6

Sep 22, 2015

The following operations can be performed on "ns acl6":

add | rm | set | unset | enable | disable | stat | rename | show

Adds an ACL6 rule to the NetScaler appliance. To commit this operation, you must apply the ACL6s. ACL6 rules filter data packets on the basis of various parameters, such as IP address, source port, action, and protocol.

add ns acl6 <acl6name> <acl6action> [-td <positive_integer>] [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-TTL <positive_integer>] [-srcMac <mac_addr>] [(-protocol <protocol> [-established]) | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-icmpType <positive_integer> [-icmpCode <positive_integer>]] [-priority <positive_integer>] [-state (ENABLED | DISABLED)]

acl6name

Name for the ACL6 rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the ACL6 rule is created.

acl6action

Action to perform on the incoming IPv6 packets that match the ACL6 rule.

Available settings function as follows:

* ALLOW - The NetScaler appliance processes the packet.

* BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.

* DENY - The NetScaler appliance drops the packet.

Possible values: BRIDGE, DENY, ALLOW

td

Traffic Domain Id.

Maximum value: 4094

srcIPv6

IP address or range of IP addresses to match against the source IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

srcPort

Port number or range of port numbers to match against the source port number of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destIPv6

IP address or range of IP addresses to match against the destination IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destPort

Port number or range of port numbers to match against the destination port number of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

TTL

Time to expire this ACL6 (in seconds).

Minimum value: 1

Maximum value: 2147483647

srcMac

MAC address to match against the source MAC address of an incoming IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an incoming IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an incoming IPv6 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance applies the ACL6 rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL6 rule to the incoming packets from all interfaces.

established

Allow only incoming TCP packets that have the ACK or RST bit set if the action set for the ACL6 rule is ALLOW and these packets match the other conditions in the ACL6 rule.

icmpType

ICMP Message type to match against the message type of an incoming IPv6 ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Maximum value: 65536

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming IPv6 ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

Maximum value: 65536

priority

Priority for the ACL6 rule, which determines the order in which it is evaluated relative to the other ACL6 rules. If you do not specify priorities while creating ACL6 rules, the ACL6 rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

state

State of the ACL6.

Possible values: ENABLED, DISABLED

Default value: XACLENABLED

```
add ns acl6 rule1 DENY -srcport 45-1024 -destIPv6 2001::45 -protocol TCP
```

Removes an ACL6 rule from the NetScaler appliance. To commit this operation, you must apply the ACL6s.

```
rm ns acl6 <acl6name> ...
```

acl6name

Name of the ACL6 rule that you want to remove.

```
rm ns acl6 rule1
```

Modifies the parameters of an ACL6 rule. To commit this operation, you must apply the ACL6s.

set ns acl6 <acl6name> [-aclaction <aclaction>] [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-icmpType <positive_integer> [-icmpCode <positive_integer>]] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-established]

acl6name

Name of the ACL6 rule whose parameters you want to modify.

aclaction

Action associated with the ACL6.

Possible values: BRIDGE, DENY, ALLOW

srcIPv6

IP address or range of IP addresses to match against the source IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

srcPort

Source Port (range).

destIPv6

IP address or range of IP addresses to match against the destination IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destPort

Destination Port (range).

srcMac

MAC address to match against the source MAC address of an incoming IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an incoming IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an incoming IPv6 packet.

Minimum value: 1

Maximum value: 255

icmpType

ICMP Message type to match against the message type of an incoming IPv6 ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Maximum value: 65536

vlan

ID of the VLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance applies the ACL6 rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL6 rule to the incoming packets from all interfaces.

priority

Priority for the ACL6 rule, which determines the order in which it is evaluated relative to the other ACL6 rules. If you do not specify priorities while creating ACL6 rules, the ACL6 rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

established

Allow only incoming TCP packets that have the ACK or RST bit set if the action set for the ACL6 rule is ALLOW and these packets match the other conditions in the ACL6 rule.

set ns acl6 rule1 -srcPort 50

Resets the attributes of the specified ACL6 rule. To commit this operation, you must apply the ACL6s.Attributes for which a default value is available revert to their default values. Refer to the set ns acl6 command for descriptions of the parameters..Refer to the set ns acl6 command for meanings of the arguments.

unset ns acl6 <acl6name> [-srcIPv6] [-srcPort] [-destIPv6] [-destPort] [-srcMac] [-protocol] [-icmpType] [-icmpCode] [-vlan] [-interface] [-established]

unset ns acl6 rule1 -srcPort

Enables an ACL6 rule. To commit this operation, you must apply the ACL6s.After you apply the ACL6 rules, the NetScaler appliance compares incoming IPv6 packets to the enabled ACL6 rules.

enable ns acl6 <acl6name> ...

acl6name

Name of ACL6 rule that you want to enable.

enable ns acl6 rule1

Disables an ACL6 rule. To commit this operation, you must apply the ACL6s.After you apply the ACL6 rules, the NetScaler appliance does not compare incoming IPv6 packets to the disabled ACL6 rules.

disable ns acl6 <acl6name> ...

acl6name

Name of ACL6 rule that you want to disable.

disable ns acl6 rule1

Displays statistics related to the ACL6 rules. To display statistics of all the ACL6 rules, run the command without any parameters. To display statistics of a particular ACL6 rule, specify the name of the ACL6 rule.

stat ns acl6 [<acl6name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

acl6name

Name of the ACL6 rule whose statistics you want the NetScaler appliance to display.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Bridge ACL6 hits (ACL6Bdg)

Packets matching a bridge IPv6 ACL, which is in transparent mode and bypasses service processing.

Deny ACL6 hits (ACL6Deny)

Packets dropped because they match IPv6 ACLs with processing mode set to DENY.

Allow ACL6 hits (ACL6Allow)

Packets matching IPv6 ACLs with processing mode set to ALLOW. NetScaler processes these packets.

NAT ACL6 hits (ACL6NAT)

Packets matching a NAT ACL6, resulting in a NAT session.

ACL6 hits (ACL6Hits)

Packets matching an IPv6 ACL.

ACL6 misses (ACL6Miss)

Packets not matching any IPv6 ACL.

NAT64 ACL6 hits (ACL6NAT64)

Packets matching a NAT64 ACL6, resulting in a NAT64 translation.

ACL6 Count (ACL6Count)

Total number of ACL6 rules configured.

Hits for this ACL6 (Hits)

Number of times the acl6 was hit

stat acl6

Renames an ACL6 rule. To commit this operation, you must apply the ACL6s.

rename ns acl6 <acl6name> <newName>

acl6name

Name of the ACL6 rule that you want to rename.

newName

New name for the ACL6 rule. Must begin with an ASCII alphabetic or underscore \\(_\\) character, and must contain only ASCII alphanumeric, underscore, hash \\(\\#\\), period \\(.\\), space, colon \\(:\\), at \\(@\\), equals \\(=\\), and hyphen \\(-\\) characters.

rename acl6 rule rule-new

Displays settings related to the ACL6 rules. To display settings of all the ACL6 rules, run the command without any parameters. To display settings of a particular ACL6 rule, specify the name of the ACL6 rule.

show ns acl6 [<acl6name>]

acl6name

Name of the ACL6 rule whose details you want the NetScaler appliance to display.

summary

fullValues

format

level

acl6action

Action to perform on the incoming IPv6 packets that match the ACL6 rule.

Available settings function as follows:

* ALLOW - The NetScaler appliance processes the packet.

* BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.

* DENY - The NetScaler appliance drops the packet.

srcMac

MAC address to match against the source MAC address of an incoming IPv6 packet.

stateflag

ACL6 state flag.

protocol

Protocol number in IPv6 header or name.

protocolNumber

Protocol number in IPv6 header or name.

srcPortVal

Source port (range).

destPortVal

Destination port (range).

srcIPv6Val

Source IPv6 address (range).

destIPv6Val

Destination IPv6 address (range).

vlan

ID of the VLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VLANs.

state

State of the ACL6.

kernelstate

Commit status of the ACL6.

TTL

Time left to expire ACL6 (in seconds).

icmpType

ICMP Message type to match against the message type of an incoming IPv6 ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming IPv6 ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

interface

ID of an interface. The NetScaler appliance applies the ACL6 rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL6 rule to the incoming packets from all interfaces.

hits

Number of hits of this ACL6.

established

This flag indicates that the ACL6 should be used for TCP response traffic only.

priority

Priority for the ACL6 rule, which determines the order in which it is evaluated relative to the other ACL6 rules. If you do not specify priorities while creating ACL6 rules, the ACL6 rules are evaluated in the order in which they are created.

operator

Logical operator.

time

Time when this acl is applied.

td

Traffic Domain Id.

devno

count

show ns acl6 rule1 1) Name: r1 Action: DENY srcIPv6 = 2001::1 destIPv6 srcMac:

ns acls

Sep 22, 2015

The following operations can be performed on "ns acls":

renumber | clear | apply

Renumbers the priorities of extended ACL rules to multiples of 10. To commit this operation, you must apply the extended ACLs. Enables you to assign a new extended ACL rule a priority that is between two existing, consecutively numbered priorities. For example, if two extended ACLs, ACL1 and ACL2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add ACL3 with priority 25.

renumber ns acls

renumber acls

Removes all simple ACL rules from the NetScaler appliance. This operation does not require an explicit apply.

clear ns acls

clear ns acls

Updates the extended ACL rule's memory tree (lookup table), adding any new extended ACL rules and applying any modifications to existing ACL rules. The lookup table includes the configuration of all the extended ACL rules on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the incoming IPv4 packets.

apply ns acls

apply ns acls

ns acls6

Sep 22, 2015

The following operations can be performed on "ns acls6":

clear | apply | renumber

Removes all simple ACL6 rules from the NetScaler appliance. This operation does not require an explicit apply.

clear ns acls6

clear ns acls6

Updates the ACL6 rules' memory tree (lookup table), adding any new ACL6 rules and applying any modifications to existing ACL rules. The lookup table includes the configuration of all the ACL6 rules on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the incoming IPv4 packets.

apply ns acls6

apply ns acls6

Renumbers the priorities of ACL6 rules to multiples of 10. To commit this operation, you must apply the ACL6s. Enables you to assign a new ACL6 rule a priority that is between two existing, consecutively numbered priorities. For example, if two ACL6s, ACL6-1 and ACL6-2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add ACL6-3 with priority 25.

renumber ns acls6

renumber acls6

---

ns appflowCollector

Sep 22, 2015

The following operations can be performed on "ns appflowCollector":

add | rm | show

Add a new AppFlow collector. NOTE: This command is deprecated.This command is deprecated in favor of 'add appflow collector'

name

Name of the AppFlow collector.

IPAddress

The IPv4 address of the AppFlow collector.

port

The UDP port on which the AppFlow collector is listening.

Default value: 4739

```
add ns appflowCollector collector1 -IPAddress 192.168.1.40 -port 2055
```

Remove an AppFlow collector. NOTE: This command is deprecated.This command is deprecated in favor of 'rm appflow collector'

name

Name of an AppFlow collector.

```
rm ns appflowCollector collector1
```

Display details of all the AppFlow collectors configured on the system. Alternatively, to view the details of a particular AppFlow collector, specify its name. NOTE: This command is deprecated.This command is deprecated in favor of 'show appflow collector'

name

Name of the AppFlow collector.

summary

fullValues

format

level

IPAddress

The IPv4 address of the AppFlow collector.

port

The UDP port on which the AppFlow collector is listening.

devno

count

stateflag

show ns appflowCollector collector1

ns appflowParam

Sep 22, 2015

The following operations can be performed on "ns appflowParam":

set | unset | show

Set AppFlow parameters. NOTE: This command is deprecated.

templateRefresh

IPFIX template refresh interval (in seconds).

Default value: 600

Minimum value: 60

Maximum value: 3600

udpPmtu

MTU to be used for IPFIX UDP packets.

Default value: 1472

Minimum value: 128

Maximum value: 1472

httpUrl

Enable AppFlow HTTP URL logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpCookie

Enable AppFlow HTTP cookie logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpReferer

Enable AppFlow HTTP referer logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpMethod

Enable AppFlow HTTP method logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpHost

Enable AppFlow HTTP host logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpUserAgent

Enable AppFlow HTTP user-agent logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientTrafficOnly

Control whether AppFlow records should be generated only for client-side traffic.

Possible values: YES, NO

Default value: NO

```
set ns appflowParam -templateRefresh 240
```

Use this command to remove ns appflowParam settings.Refer to the set ns appflowParam command for meanings of the arguments.NOTE: This command is deprecated.

Display AppFlow parameters. NOTE: This command is deprecated.This command is deprecated in favor of 'show appflow param'

summary

fullValues

format

level

templateRefresh

IPFIX template refresh interval (in seconds).

udpPmtu

MTU to be used for IPFIX UDP packets.

httpUrl

Enable AppFlow HTTP URL logging.

httpCookie

Enable AppFlow HTTP cookie logging.

httpReferer

Enable AppFlow HTTP referer logging.

httpMethod

Enable AppFlow HTTP method logging.

httpHost

Enable AppFlow HTTP host logging.

httpUserAgent

Enable AppFlow HTTP user-agent logging.

clientTrafficOnly

Control whether AppFlow records should be generated only for client-side traffic.

ns aptlicense

Sep 22, 2015

The following operations can be performed on "ns aptlicense":

show | update

show ns aptlicense <serialNo>

serialNo

Hardware Serial Number/License Activation Code(LAC)

response

Response data as text blob

id

License ID

sessionId

Session ID

bindType

Bind type

countAvailable

Count

countTotal

Count

name

License name

relevance

License relevance

datePurchased

License purchase date

dateSa

License SA date

dateExp

License expiry date

features

Features

show ns aptlicense <hw-no/lac>

---

update ns aptlicense <id> <sessionId> <bindType> <countAvailable> [<licenseDir>]

id

License ID

sessionId

Session ID

bindType

Bind type

countAvailable

Count

licenseDir

License Directory

update ns aptlicense key1 sessionID# HOSTNAME 1

---

ns config

Sep 22, 2015

The following operations can be performed on "ns config":

clear | set | unset | save | show | diff

Clears the NetScaler running configurations based on different levels.

clear ns config [-force] <level>

force

Configurations will be cleared without prompting for confirmation.

level

Types of configurations to be cleared.

* basic: Clears all configurations except the following:

- NSIP, default route (gateway), MIPs, and SNIPs

- Network settings (DG, VLAN, RHI, NTP and DNS settings)

- Cluster settings

- HA node definitions

- Feature and mode settings

- nsroot password

* extended: Clears the same configurations as the 'basic' option. In addition, it clears the nsroot password and feature and mode settings.

* full: Clears all configurations except NSIP, default route, and interface settings.

Note: When you clear the configurations through the cluster IP address, by specifying the level as 'full', the cluster is deleted and all cluster nodes become standalone appliances. The 'basic' and 'extended' levels are propagated to the cluster nodes.

Possible values: basic, extended, full

Sets the NetScaler IP address and NetScaler VLAN. To set other NetScaler parameters, use the 'set ns param' command. Note: To change the NSIP address or the NSVLAN of an appliance that is part of a cluster, first remove the appliance from the cluster, change the NSIP or the NSVLAN, and then add the appliance back to the cluster.

set ns config [-IPAddress <ip_addr> -netmask <netmask>] [-nsvlan <positive_integer> -ifnum <interface_name> ... [-tagged (YES | NO)]] [-nwfwmode <nwfwmode>]

IPAddress

IP address of the NetScaler appliance. Commonly referred to as NSIP address. This parameter is mandatory to bring up the appliance.

nsvlan

VLAN (NSVLAN) for the subnet on which the IP address resides.

Minimum value: 2

Maximum value: 4094

httpPort

The HTTP ports on the Web server. This allows the system to perform connection off-load for any client request that has a destination port matching one of these configured ports.

Minimum value: 1

maxConn

The maximum number of connections that will be made from the system to the web server(s) attached to it. The value entered here is applied globally to all attached servers.

Maximum value: 4294967294

maxReq

The maximum number of requests that the system can pass on a particular connection between the system and a server attached to it. Setting this value to 0 allows an unlimited number of

requests to be passed.

Maximum value: 65535

cip

The option to control (enable or disable) the insertion of the actual client IP address into the HTTP header request passed from the client to one, some, or all servers attached to the system.

The passed address can then be accessed through a minor modification to the server.

l If cipHeader is specified, it will be used as the client IP header.

l If it is not specified, then the value that has been set by the set ns config CLI command will be used as the client IP header.

Possible values: ENABLED, DISABLED

cookieversion

The version of the cookie inserted by system.

Possible values: 0, 1

secureCookie

enable/disable secure flag for persistence cookie

Possible values: ENABLED, DISABLED

Default value: ENABLED

pmtuMin

The minimum Path MTU.

Default value: 576

Minimum value: 168

Maximum value: 1500

pmtuTimeout

The timeout value in minutes.

Default value: 10

Minimum value: 1

Maximum value: 1440

ftpPortRange

Port range configured for FTP services.

Minimum value: 1024

Maximum value: 64000

crPortRange

Port range for cache redirection services.

Minimum value: 1

Maximum value: 65535

timezone

Name of the timezone

Possible values: CoordinatedUniversalTime, GMT+01:00-CET-Europe/Andorra, GMT+04:00-GST-Asia/Dubai, GMT+04:30-AFT-Asia/Kabul, GMT-04:00-AST-America/Antigua, GMT-04:00-AST-America/Anguilla, GMT+01:00-CET-Europe/Tirane, GMT+04:00-AMT-Asia/Yerevan, GMT+01:00-WAT-Africa/Luanda, GMT+13:00-NZDT-Antarctica/McMurdo, GMT+13:00-NZDT-Antarctica/South_Pole, GMT-03:00-ROTT-Antarctica/Rothera, GMT-04:00-CLT-Antarctica/Palmer, GMT+05:00-MAWT-Antarctica/Mawson, GMT+07:00-DAVT-Antarctica/Davis, GMT+08:00-WST-Antarctica/Casey, GMT+06:00-VOST-Antarctica/Vostok, GMT+10:00-DDUT-Antarctica/DumontDUrville, GMT+03:00-SYOT-Antarctica/Syowa, GMT+11:00-MIST-Antarctica/Macquarie, GMT-03:00-ART-America/Argentina/Buenos_Aires, GMT-03:00-ART-America/Argentina/Cordoba, GMT-03:00-ART-America/Argentina/Salta, GMT-03:00-ART-America/Argentina/Jujuy, GMT-03:00-ART-America/Argentina/Tucuman, GMT-03:00-ART-America/Argentina/Catamarca, GMT-03:00-ART-America/Argentina/La_Rioja, GMT-03:00-ART-America/Argentina/San_Juan, GMT-03:00-ART-America/Argentina/Mendoza, GMT-03:00-WARST-America/Argentina/San_Luis, GMT-03:00-ART-America/Argentina/Rio_Gallegos, GMT-03:00-ART-America/Argentina/Ushuaia, GMT-11:00-SST-Pacific/Pago_Pago, GMT+01:00-CET-Europe/Vienna, GMT+11:00-LHST-Australia/Lord_Howe, GMT+11:00-EST-Australia/Hobart, GMT+11:00-EST-Australia/Currie, GMT+11:00-EST-Australia/Melbourne, GMT+11:00-EST-Australia/Sydney, GMT+10:30-CST-Australia/Broken_Hill, GMT+10:00-EST-Australia/Brisbane, GMT+10:00-EST-Australia/Lindeman, GMT+10:30-CST-Australia/Adelaide, GMT+09:30-CST-Australia/Darwin, GMT+08:00-WST-Australia/Perth, GMT+08:45-CWST-Australia/Eucla, GMT-04:00-AST-America/Aruba, GMT+02:00-EET-Europe/Mariehamn, GMT+04:00-AZT-Asia/Baku, GMT+01:00-CET-Europe/Sarajevo, GMT-04:00-AST-America/Barbados, GMT+06:00-BDT-Asia/Dhaka, GMT+01:00-CET-Europe/Brussels, GMT+00:00-GMT-Africa/Ouagadougou, GMT+02:00-EET-Europe/Sofia, GMT+03:00-AST-Asia/Bahrain, GMT+02:00-CAT-Africa/Bujumbura, GMT+01:00-WAT-Africa/Porto-Novo, GMT-04:00-AST-America/St_Barthelemy, GMT-03:00-ADT-Atlantic/Bermuda, GMT+08:00-BNT-Asia/Brunei, GMT-04:00-BOT-America/La_Paz, GMT-02:00-FNT-America/Noronha, GMT-03:00-BRT-America/Belem, GMT-03:00-BRT-America/Fortaleza, GMT-03:00-BRT-America/Recife, GMT-03:00-BRT-America/Araguaina, GMT-03:00-BRT-America/Maceio, GMT-03:00-BRT-America/Bahia, GMT-03:00-BRT-America/Sao_Paulo, GMT-04:00-AMT-America/Campo_Grande, GMT-04:00-AMT-America/Cuiaba, GMT-03:00-BRT-America/Santarem, GMT-04:00-AMT-America/Porto_Velho, GMT-04:00-AMT-America/Boa_Vista, GMT-04:00-AMT-America/Manaus, GMT-04:00-AMT-America/Eirunepe, GMT-04:00-AMT-America/Rio_Branco, GMT-04:00-EDT-America/Nassau, GMT+06:00-BTT-Asia/Thimphu, GMT+02:00-CAT-Africa/Gaborone, GMT+03:00-FET-Europe/Minsk, GMT-06:00-CST-America/Belize,

GMT-02:30-NDT-America/St_Johns, GMT-03:00-ADT-America/Halifax, GMT-03:00-ADT-America/Glace_Bay, GMT-03:00-ADT-America/Moncton, GMT-03:00-ADT-America/Goose_Bay, GMT-04:00-AST-America/Blanc-Sablon, GMT-04:00-EDT-America/Montreal, GMT-04:00-EDT-America/Toronto, GMT-04:00-EDT-America/Nipigon, GMT-04:00-EDT-America/Thunder_Bay, GMT-04:00-EDT-America/Iqaluit, GMT-04:00-EDT-America/Pangnirtung, GMT-05:00-CDT-America/Resolute, GMT-05:00-EST-America/Atikokan, GMT-05:00-CDT-America/Rankin_Inlet, GMT-05:00-CDT-America/Winnipeg, GMT-05:00-CDT-America/Rainy_River, GMT-06:00-CST-America/Regina, GMT-06:00-CST-America/Swift_Current, GMT-06:00-MDT-America/Edmonton, GMT-06:00-MDT-America/Cambridge_Bay, GMT-06:00-MDT-America/Yellowknife, GMT-06:00-MDT-America/Inuvik, GMT-07:00-MST-America/Dawson_Creek, GMT-07:00-PDT-America/Vancouver, GMT-07:00-PDT-America/Whitehorse, GMT-07:00-PDT-America/Dawson, GMT+06:30-CCT-Indian/Cocos, GMT+01:00-WAT-Africa/Kinshasa, GMT+02:00-CAT-Africa/Lubumbashi, GMT+01:00-WAT-Africa/Bangui, GMT+01:00-WAT-Africa/Brazzaville, GMT+01:00-CET-Europe/Zurich, GMT+00:00-GMT-Africa/Abidjan, GMT-10:00-CKT-Pacific/Rarotonga, GMT-04:00-CLT-America/Santiago, GMT-06:00-EAST-Pacific/Easter, GMT+01:00-WAT-Africa/Douala, GMT+08:00-CST-Asia/Shanghai, GMT+08:00-CST-Asia/Harbin, GMT+08:00-CST-Asia/Chongqing, GMT+08:00-CST-Asia/Urumqi, GMT+08:00-CST-Asia/Kashgar, GMT-05:00-COT-America/Bogota, GMT-06:00-CST-America/Costa_Rica, GMT-04:00-CDT-America/Havana, GMT-01:00-CVT-Atlantic/Cape_Verde, GMT+07:00-CXT-Indian/Christmas, GMT+02:00-EET-Asia/Nicosia, GMT+01:00-CET-Europe/Prague, GMT+01:00-CET-Europe/Berlin, GMT+03:00-EAT-Africa/Djibouti, GMT+01:00-CET-Europe/Copenhagen, GMT-04:00-AST-America/Dominica, GMT-04:00-AST-America/Santo_Domingo, GMT+01:00-CET-Africa/Algiers, GMT-05:00-ECT-America/Guayaquil, GMT-06:00-GALT-Pacific/Galapagos, GMT+02:00-EET-Europe/Tallinn, GMT+02:00-EET-Africa/Cairo, GMT+00:00-WET-Africa/El_Aaiun, GMT+03:00-EAT-Africa/Asmara, GMT+01:00-CET-Europe/Madrid, GMT+01:00-CET-Africa/Ceuta, GMT+00:00-WET-Atlantic/Canary, GMT+03:00-EAT-Africa/Addis_Ababa, GMT+02:00-EET-Europe/Helsinki, GMT+12:00-FJT-Pacific/Fiji, GMT-03:00-FKST-Atlantic/Stanley, GMT+10:00-CHUT-Pacific/Chuuk, GMT+11:00-PONT-Pacific/Pohnpei, GMT+11:00-KOST-Pacific/Kosrae, GMT+00:00-WET-Atlantic/Faroe, GMT+01:00-CET-Europe/Paris, GMT+01:00-WAT-Africa/Libreville, GMT+00:00-GMT-Europe/London, GMT-04:00-AST-America/Grenada, GMT+04:00-GET-Asia/Tbilisi, GMT-03:00-GFT-America/Cayenne, GMT+00:00-GMT-Europe/Guernsey, GMT+00:00-GMT-Africa/Accra, GMT+01:00-CET-Europe/Gibraltar, GMT-03:00-WGT-America/Godthab, GMT+00:00-GMT-America/Danmarkshavn, GMT-01:00-EGT-America/Scoresbysund, GMT-03:00-ADT-America/Thule, GMT+00:00-GMT-Africa/Banjul, GMT+00:00-GMT-Africa/Conakry, GMT-04:00-AST-America/Guadeloupe, GMT+01:00-WAT-Africa/Malabo, GMT+02:00-EET-Europe/Athens, GMT-02:00-GST-Atlantic/South_Georgia, GMT-06:00-CST-America/Guatemala, GMT+10:00-ChST-Pacific/Guam, GMT+00:00-GMT-Africa/Bissau, GMT-04:00-GYT-America/Guyana, GMT+08:00-HKT-Asia/Hong_Kong, GMT-06:00-CST-America/Tegucigalpa, GMT+01:00-CET-Europe/Zagreb, GMT-05:00-EST-America/Port-au-Prince, GMT+01:00-CET-Europe/Budapest, GMT+07:00-WIT-Asia/Jakarta, GMT+07:00-WIT-Asia/Pontianak, GMT+08:00-CIT-Asia/Makassar, GMT+09:00-EIT-Asia/Jayapura, GMT+00:00-GMT-Europe/Dublin, GMT+02:00-IST-Asia/Jerusalem, GMT+00:00-GMT-Europe/Isle_of_Man, GMT+05:30-IST-Asia/Kolkata, GMT+06:00-IOT-Indian/Chagos, GMT+03:00-AST-Asia/Baghdad, GMT+03:30-IRST-Asia/Tehran, GMT+00:00-GMT-Atlantic/Reykjavik, GMT+01:00-CET-Europe/Rome, GMT+00:00-GMT-Europe/Jersey, GMT-05:00-EST-America/Jamaica, GMT+02:00-EET-Asia/Amman, GMT+09:00-JST-Asia/Tokyo, GMT+03:00-EAT-Africa/Nairobi, GMT+06:00-KGT-Asia/Bishkek, GMT+07:00-ICT-Asia/Phnom_Penh, GMT+12:00-GILT-Pacific/Tarawa, GMT+13:00-PHOT-Pacific/Enderbury, GMT+14:00-LINT-Pacific/Kiritimati, GMT+03:00-EAT-Indian/Comoro, GMT-04:00-AST-America/St_Kitts, GMT+09:00-KST-Asia/Pyongyang, GMT+09:00-KST-Asia/Seoul, GMT+03:00-AST-Asia/Kuwait, GMT-05:00-EST-America/Cayman, GMT+06:00-ALMT-Asia/Almaty, GMT+06:00-QYZT-Asia/Qyzylorda, GMT+05:00-AQTT-Asia/Aqtobe, GMT+05:00-AQTT-Asia/Aqtau, GMT+05:00-ORAT-Asia/Oral, GMT+07:00-ICT-Asia/Vientiane, GMT+02:00-EET-Asia/Beirut, GMT-04:00-AST-America/St_Lucia, GMT+01:00-CET-Europe/Vaduz, GMT+05:30-IST-Asia/Colombo, GMT+00:00-GMT-Africa/Monrovia, GMT+02:00-SAST-Africa/Maseru, GMT+02:00-EET-Europe/Vilnius, GMT+01:00-CET-Europe/Luxembourg, GMT+02:00-EET-Europe/Riga, GMT+02:00-EET-Africa/Tripoli, GMT+00:00-WET-Africa/Casablanca, GMT+01:00-CET-Europe/Monaco, GMT+02:00-EET-Europe/Chisinau, GMT+01:00-CET-Europe/Podgorica, GMT-04:00-AST-America/Marigot, GMT+03:00-EAT-Indian/Antananarivo, GMT+12:00-MHT-Pacific/Majuro, GMT+12:00-MHT-Pacific/Kwajalein, GMT+01:00-CET-Europe/Skopje, GMT+00:00-GMT-Africa/Bamako, GMT+06:30-MMT-Asia/Rangoon, GMT+08:00-ULAT-Asia/Ulaanbaatar, GMT+07:00-HOVT-Asia/Hovd, GMT+08:00-CHOT-Asia/Choibalsan, GMT+08:00-CST-Asia/Macau, GMT+10:00-ChST-Pacific/Saipan, GMT-04:00-AST-America/Martinique, GMT+00:00-GMT-Africa/Nouakchott, GMT-04:00-AST-America/Montserrat, GMT+01:00-CET-Europe/Malta, GMT+04:00-MUT-Indian/Mauritius, GMT+05:00-MVT-Indian/Maldives, GMT+02:00-CAT-Africa/Blantyre, GMT-06:00-CST-America/Mexico_City, GMT-06:00-CST-America/Cancun, GMT-06:00-CST-America/Merida, GMT-06:00-CST-America/Monterrey, GMT-05:00-CDT-America/Matamoros, GMT-07:00-MST-America/Mazatlan, GMT-07:00-MST-America/Chihuahua, GMT-06:00-MDT-America/Ojinaga, GMT-07:00-MST-America/Hermosillo, GMT-07:00-PDT-America/Tijuana, GMT-08:00-PST-America/Santa_Isabel, GMT-06:00-CST-America/Bahia_Banderas, GMT+08:00-MYT-Asia/Kuala_Lumpur, GMT+08:00-MYT-Asia/Kuching, GMT+02:00-CAT-Africa/Maputo, GMT+02:00-WAST-Africa/Windhoek, GMT+11:00-NCT-Pacific/Noumea, GMT+01:00-WAT-Africa/Niamey, GMT+11:30-NFT-Pacific/Norfolk, GMT+01:00-WAT-Africa/Lagos, GMT-06:00-CST-America/Managua, GMT+01:00-CET-Europe/Amsterdam, GMT+01:00-CET-Europe/Oslo, GMT+05:45-NPT-Asia/Kathmandu, GMT+12:00-NRT-Pacific/Nauru, GMT-11:00-NUT-Pacific/Niue, GMT+13:00-NZDT-Pacific/Auckland, GMT+13:45-CHADT-Pacific/Chatham, GMT+04:00-GST-Asia/Muscat, GMT-05:00-EST-America/Panama, GMT-05:00-PET-America/Lima, GMT-10:00-TAHT-Pacific/Tahiti, GMT-09:30-MART-Pacific/Marquesas, GMT-09:00-GAMT-Pacific/Gambier, GMT+10:00-PGT-Pacific/Port_Moresby, GMT+08:00-PHT-Asia/Manila, GMT+05:00-PKT-Asia/Karachi, GMT+01:00-CET-Europe/Warsaw, GMT-02:00-PMDT-America/Miquelon, GMT-08:00-PST-Pacific/Pitcairn, GMT-04:00-AST-America/Puerto_Rico, GMT+02:00-EET-Asia/Gaza, GMT+02:00-EET-Asia/Hebron, GMT+00:00-WET-Europe/Lisbon, GMT+00:00-WET-Atlantic/Madeira, GMT-01:00-AZOT-Atlantic/Azores, GMT+09:00-PWT-Pacific/Palau, GMT-03:00-PYST-America/Asuncion, GMT+03:00-AST-Asia/Qatar, GMT+04:00-RET-Indian/Reunion, GMT+02:00-EET-Europe/Bucharest, GMT+01:00-CET-Europe/Belgrade, GMT+03:00-FET-Europe/Kaliningrad, GMT+04:00-MSK-Europe/Moscow, GMT+04:00-VOLT-Europe/Volgograd, GMT+04:00-SAMT-Europe/Samara, GMT+06:00-YEKT-Asia/Yekaterinburg, GMT+07:00-OMST-Asia/Omsk, GMT+07:00-NOVT-Asia/Novosibirsk, GMT+07:00-NOVT-Asia/Novokuznetsk, GMT+08:00-KRAT-Asia/Krasnoyarsk, GMT+09:00-IRKT-Asia/Irkutsk, GMT+10:00-YAKT-Asia/Yakutsk, GMT+11:00-VLAT-Asia/Vladivostok, GMT+11:00-SAKT-Asia/Sakhalin, GMT+12:00-MAGT-Asia/Magadan, GMT+12:00-PETT-Asia/Kamchatka, GMT+12:00-ANAT-Asia/Anadyr, GMT+02:00-CAT-Africa/Kigali, GMT+03:00-AST-Asia/Riyadh, GMT+11:00-SBT-Pacific/Guadalcanal, GMT+04:00-SCT-Indian/Mahe, GMT+03:00-EAT-Africa/Khartoum, GMT+01:00-CET-Europe/Stockholm, GMT+08:00-SGT-Asia/Singapore, GMT+00:00-GMT-Atlantic/St_Helena, GMT+01:00-CET-Europe/Ljubljana, GMT+01:00-CET-Arctic/Longyearbyen, GMT+01:00-CET-Europe/Bratislava, GMT+00:00-GMT-Africa/Freetown, GMT+01:00-CET-Europe/San_Marino, GMT+00:00-GMT-Africa/Dakar, GMT+03:00-EAT-Africa/Mogadishu, GMT-03:00-SRT-America/Paramaribo, GMT+00:00-GMT-Africa/Sao_Tome, GMT-06:00-CST-America/El_Salvador, GMT+02:00-EET-Asia/Damascus, GMT+02:00-SAST-Africa/Mbabane, GMT-04:00-EDT-America/Grand_Turk, GMT+01:00-WAT-Africa/Ndjamena, GMT+05:00-TFT-Indian/Kerguelen, GMT+00:00-GMT-Africa/Lome, GMT+07:00-ICT-Asia/Bangkok, GMT+05:00-TJT-Asia/Dushanbe, GMT-10:00-TKT-Pacific/Fakaofo, GMT+09:00-TLT-Asia/Dili, GMT+05:00-TMT-Asia/Ashgabat, GMT+01:00-CET-Africa/Tunis, GMT+13:00-TOT-Pacific/Tongatapu, GMT+02:00-EET-Europe/Istanbul, GMT-04:00-AST-America/Port_of_Spain, GMT+12:00-TVT-Pacific/Funafuti, GMT+08:00-CST-Asia/Taipei, GMT+03:00-EAT-Africa/Dar_es_Salaam, GMT+02:00-EET-Europe/Kiev, GMT+02:00-EET-Europe/Uzhgorod, GMT+02:00-EET-Europe/Zaporozhye, GMT+02:00-EET-Europe/Simferopol, GMT+03:00-EAT-Africa/Kampala, GMT-10:00-HST-Pacific/Johnston, GMT-11:00-SST-Pacific/Midway, GMT+12:00-WAKT-Pacific/Wake, GMT-04:00-EDT-America/New_York, GMT-04:00-EDT-America/Detroit, GMT-04:00-EDT-America/Kentucky/Louisville, GMT-04:00-EDT-America/Kentucky/Monticello, GMT-04:00-EDT-America/Indiana/Indianapolis, GMT-04:00-EDT-America/Indiana/Vincennes, GMT-04:00-EDT-America/Indiana/Winamac, GMT-04:00-EDT-America/Indiana/Marengo, GMT-04:00-EDT-America/Indiana/Petersburg, GMT-04:00-EDT-America/Indiana/Vevay, GMT-05:00-CDT-America/Chicago, GMT-05:00-CDT-America/Indiana/Tell_City, GMT-05:00-CDT-America/Indiana/Knox, GMT-05:00-CDT-America/Menominee, GMT-05:00-CDT-America/North_Dakota/Center, GMT-05:00-CDT-America/North_Dakota/New_Salem, GMT-05:00-CDT-America/North_Dakota/Beulah, GMT-06:00-MDT-America/Denver, GMT-06:00-MDT-America/Boise, GMT-06:00-MDT-America/Shiprock, GMT-07:00-MST-America/Phoenix, GMT-07:00-PDT-America/Los_Angeles, GMT-08:00-AKDT-America/Anchorage, GMT-08:00-AKDT-America/Juneau, GMT-08:00-AKDT-America/Sitka, GMT-08:00-AKDT-America/Yakutat, GMT-08:00-AKDT-America/Nome, GMT-09:00-HADT-America/Adak, GMT-08:00-MeST-America/Metlakatla, GMT-10:00-HST-Pacific/Honolulu, GMT-03:00-UYT-America/Montevideo, GMT+05:00-UZT-Asia/Samarkand, GMT+05:00-UZT-Asia/Tashkent, GMT+01:00-CET-Europe/Vatican, GMT-04:00-AST-America/St_Vincent, GMT-04:30-VET-America/Caracas, GMT-04:00-AST-America/Tortola, GMT-04:00-AST-America/St_Thomas, GMT+07:00-ICT-Asia/Ho_Chi_Minh, GMT+11:00-VUT-Pacific/Efate, GMT+12:00-WFT-Pacific/Wallis, GMT+14:00-WSDT-Pacific/Apia, GMT+03:00-AST-Asia/Aden, GMT+03:00-EAT-Indian/Mayotte, GMT+02:00-SAST-Africa/Johannesburg, GMT+02:00-CAT-Africa/Lusaka, GMT+02:00-CAT-Africa/Harare

grantQuotaMaxClient

The percentage of shared quota to be granted at a time for maxClient

Default value: 10

Maximum value: 100

exclusiveQuotaMaxClient

The percentage of maxClient to be given to PEs

Default value: 80

Maximum value: 100

grantQuotaSpillOver

The percentage of shared quota to be granted at a time for spillover

Default value: 10

Maximum value: 100

exclusiveQuotaSpillOver

The percentage of max limit to be given to PEs

Default value: 80

Maximum value: 100

nwfwmode

Network Firewall mode to be used.

NOFIREWALL - No Network firewall setting

BASIC - DENY-ALL behavior and DENY-ALL AT BOOTUP

EXTENDED - NS_NWFWMODE_BASIC + drop IP fragments + TCP and ACL logging + packet drop on closed port

EXTENDEDPLUS - NS_NWFWMODE_EXTENDED + block traffic on 3008-3011 + drop non-session packets

FULL - NS_NWFWMODE_EXTENDEDPLUS + drop non-ip packets.

Possible values: NOFIREWALL, BASIC, EXTENDED, EXTENDEDPLUS, FULL

Default value: NS_NWFWMODE_NO

Removes the attributes of the NetScaler appliance. Attributes for which a default value is available revert to their default values. Refer to the 'set ns config' command for a description of the parameters..Refer to the set ns config command for meanings of the arguments.

unset ns config [-nsvlan] [-IPAddress] [-netmask] [-ifnum] [-tagged] [-nwfwmode]

Save the configurations to the appliances FLASH memory in the /nsconfig/ns.conf file. Backup configuration files are named ns.conf.n. The most recent backup file has the smallest value for n.

save ns config

message

Displays the following details of the NetScaler appliance: * NetScaler IP address and subnet mask * Number of mapped IP addresses * Identifies the appliance as a standalone appliance, a part of a HA pair, or is a cluster node * Current time on the system and timestamp when the appliance was last updated Note: To view the complete configurations that have been executed on the appliance, run the 'show ns runningConfig' command.

show ns config

format

level

IPAddress

IP Address of the System.

netmask

The netmask corresponding to the IP address.

mappedIP

Mapped IP Address of the System.

range

The range of Mapped IP addresses to be configured.

nsvlan

The VLAN (NSVLAN) for the subnet on which the system IP resides.

ifnum

Bind the given ports to the NSVLAN.

tagged

Specifies that the interfaces will be added as 802.1q tagged interfaces. Packets sent on these interface on this VLAN will have an additional 4-byte 802.1q tag which identifies the VLAN.

To use 802.1q tagging, the switch connected to the appliance's interfaces must also be configured for tagging.

httpPort

The HTTP ports on the Web server.

maxConn

Maximum Number of Connections.

maxReq

Maxmimum Number of requests that can be handled.

cip

Insertion of client IP address into the HTTP header.

cipHeader

The text that will be used as the client IP header.

cookieversion

The version of the cookie inserted by system.

secureCookie

enable/disable secure flag for persistence cookie

failover

Standalone node.NOTE: This attribute is deprecated.

systemType

The type of the System. Possible Values: Standalone, HA, Cluster

primaryIP

HA Master Node IP address.

pmtuMin

The minimum Path MTU.

pmtuTimeout

The timeout value in minutes.

ftpPortRange

Port range configured for FTP services.

crPortRange

Port range for cache redirection services.

flags

The flags for this entry.

timezone

Name of the timezone

LastConfigChangedTime

Time when the configuration was last modified.

LastConfigSaveTime

Time when the configuration was last saved through savensconfig.

currentSytemTime

current system time in date format.

systemTime

current system time.

grantQuotaMaxClient

The percentage of shared quota to be granted at a time for maxClient

exclusiveQuotaMaxClient

The percentage of maxClient to be given to PEs

grantQuotaSpillOver

The percentage of shared quota to be granted at a time for spillover

exclusiveQuotaSpillOver

The percentage of max limit to be given to PEs

nwfwmode

Network Firewallmode

Difference between two configuration

diff ns config [<config1>] [<config2>] [-outtype (cli | xml)] [-template] [-ignoreDeviceSpecific]

config1

Location of the configurations.

config2

Location of the configurations.

outtype

Format to display the difference in configurations.

Possible values: cli, xml

template

File that contains the commands to be compared.

ignoreDeviceSpecific

Suppress device specific differences.

response

Generates the differences between two configurations. Note: If no parameters are provided, then the differences between the saved configurations and the runnin

---

ns connectiontable

Sep 22, 2015

The following operations can be performed on "ns connectiontable":

Displays the current TCP/IP connection table.

show ns connectiontable [<filterexpression>] [-detail <detail> ...]

filterexpression

Expression using which TCP/IP connections must be filtered. Maximum length of filter is 255 and it can be of the following format:

"<expression> \\[<relop> <expression>"\\]

where,

<relop> can be the && or the || relational operators.

<expression> is a string in the following format:

<qualifier> <operator> <qualifier-value>

where,

<operator> can be any one of the following (except the commas): ==, eq, !=, neq, >, gt, <, lt, >=, ge, <=, le, BETWEEN

Following are the valid qualifier - qualifier value combinations:

SOURCEIP - A valid IP address

SOURCEPORT - A valid port number

DESTIP - A valid IP address

DESTPORT - A valid port number

IP - A valid IP address

PORT - A valid port number

IDLETIME - A positive integer indicating the idle time

SVCNAME - The name of a service

VSVRNAME - The name of a vserver

STATE - CLOSE_WAIT, CLOSED, CLOSING, ESTABLISHED, FIN_WAIT_1, FIN_WAIT_2, LAST_ACK, LISTEN, SYN_RECEIVED, SYN_SENT, TIME_WAIT

SVCTYPE - HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, NNTP, RPCSVR, RPCSVRS, RPCCLNT, DNS, ADNS, SNMP, RTSP, DHCPRA, ANY, MONITOR, MONITOR_UDP, MONITOR_PING, SIP_UDP, MYSQL, MSSQL, UNKNOWN.

link

Display link information if available

name

Display name instead of IP for local entities

detail

Specify display options for the connection table.

* LINK - Displays the linked PCB (Protocol Control Block).

* NAME - Displays along with the service name.

* CONNFAILOVER - Displays PCB with connection failover.

* FULL - Displays all available details.

summary

fullValues

SOURCEIP

Source IP of the connection.

SOURCEPORT

Source port of the connection.

DESTIP

Destination IP of the connection.

DESTPORT

Destination port of the connection.

SVCTYPE

Protocol supported by the connection.

IDLETIME

Time since last activity was detected on the connection.

STATE

Current TCP/IP state of the connection.

linkSourceIP

Source IP of the link connection.

linkSourcePort

Source port of the link connection.

linkDestIP

Destination IP of the link connection.

linkDestPort

Destination port of the link connection.

linkServiceType

Protocol supported by the link connection.

linkIdleTime

Time since last activity was detected on link connection.

linkState

TCP/IP current state of link connection.

entityName

NetScaler entity name for the connection.

linkEntityName

NetScaler entity name for link connection.

connectionNumber

Connection numberNOTE: This attribute is deprecated.Deprecated in favour of NSA_CONNID.

linkConnectionNumber

Link connection numberNOTE: This attribute is deprecated.Deprecated in favour of NSA_LINK_CONNID.

connid

Unique transaction number for the connection.

linkConnid

Unique transaction number for the peer connection.

filterFlags

flags used to store display options

optionFlags

flags used to store TCP options like Sack, WS

nsWSvalue

netscaler window scaling value

peerWSvalue

peer window scaling value

mss

Client side MSS for the connection - used in server SYN.

retxRetryCnt

Retransmission retry count for the connection.

rcvWnd

Received Advertised Window for the connection.

advWnd

Sent advertised window for the connection.

sndCwnd

sent congestion window for the connection.

iss

Initial send sequence number for the connection.

irs

Initial receive sequence number for the connection.

rcvNxt

next expecting seq number for the connection.

maxAck

current running max ack sent for the connection.

sndNxt

next bytes seq number for the connection.

sndUnAck

Most recently received ACK for the connection.

httpEndSeq

HTTP parsing tracking seq number for the connection.

httpState

HTTP Protocol state for the connection.

trCount

Max reuests allowed per connection.

priority

priority of the connection.

httpReqVer

current HTTP request version on the connection.

httpRequest

current HTTP request type on the connection.

httpRspCode

current response type on the connection.

rttSmoothed

smoothed RTT value of the connection.

rttVariance

RTT variance for the connection.

outoforderPkts

held packets on the connection.

count

linkOptionFlag

Link connection's TCP option flag for Sack and WS

linknsWSvalue

Link connection-s netscaler window scaling value

linkpeerWSvalue

Link connection-s peer netscaler window scaling value

linkMSS

Client side MSS for the Link connection - used in server SYN

linkRetxRetryCnt

Retransmission retry count for the Link connection.

linkRcvWnd

Received Advertised Window for the Link connection.

linkAdvWnd

Sent advertised window for the Link connection.

linkSndCwnd

Send congestion window for the Link connection.

linkISS

Initial send seq number for the Link connection.

linkIRS

Initial receive seq number for the Link connection.

linkRcvNxt

Next expecting seq number on the Link connection.

linkMaxAck

Current running maximum ack sent on the Link connection.

linkSndNxt

Next bytes seq number for the Link connection.

linkSndUnAck

Most recently received ACK on the Link connection.

linkHttpEndSeq

HTTP parsing tracking seq number on the Link connection.

linkHttpState

HTTP protocol state on the Link connection.

linkTrCount

Max requests per connection for Link connection.

linkPriority

Priority for the Link connection.

linkHttpReqVer

HTTP current request version on Link connection.

linkHttpReq

HTTP current request type on Link connection.

linkHttpRspCode

Current response type on link connection.

linkRttSmoothed

Smoothed RTT value on link connection.

linkRttVariance

RTT variance on Link connection.

linkHeldPkts

Held packets on Link connection.

targetnodeidnnm

NNM connection target node ID.

sourcenodeidnnm

NNM connection source node ID.

channelidnnm

NNM connection channel ID.

msgversionnnm

nnm message version.

td

Traffic Domain Id.

devno

stateflag

ns consoleloginprompt

Sep 22, 2015

The following operations can be performed on "ns consoleloginprompt":

set | unset | show

set ns consoleloginprompt <promptString>

promptString

Console login prompt string

set ns consoleloginprompt <prompt_string>

Use this command to remove ns consoleloginprompt settings.Refer to the set ns consoleloginprompt command for meanings of the arguments.

unset ns consoleloginprompt -promptString

---

show ns consoleloginprompt

promptString

Console login prompt string

get ns consoleloginprompt

---

ns dhcpIp

Sep 22, 2015

The following operations can be performed on "ns dhcpIp":

Releases the IP address acquired by the DHCP client.

release ns dhcpIp

ns dhcpParams

Sep 22, 2015

The following operations can be performed on "ns dhcpParams":

set | unset | show

Sets the DHCP client parameters.

```
set ns dhcpParams [-dhcpClient ( ON | OFF )] [-saveroute ( ON | OFF )]
```

dhcpClient

Enables DHCP client to acquire IP address from the DHCP server in the next boot. When set to OFF, disables the DHCP client in the next boot.

Possible values: ON, OFF

Default value: OFF

saveroute

DHCP acquired routes are saved on the NetScaler appliance.

Possible values: ON, OFF

Default value: OFF

Use this command to remove ns dhcpParams settings.Refer to the set ns dhcpParams command for meanings of the arguments.

```
unset ns dhcpParams [-dhcpClient] [-saveroute]
```

Displays the parameters configured for the DHCP client.

```
show ns dhcpParams
```

format

level

dhcpClient

ON, if DHCP client active on next reboot, else OFF

IPAddress

DHCP acquired IP

netmask

DHCP acquired Netmask

hostRtGw

DHCP acquired Gateway

running

DHCP mode

saveroute

DHCP acquired gateway save flag

ns diameter

Sep 22, 2015

The following operations can be performed on "ns diameter":

set | unset | show

Set the diameter configuration on NS.

set ns diameter [-identity <string>] [-realm <string>] [-serverClosePropagation (YES | NO)]

identity

DiameterIdentity to be used by NS. DiameterIdentity is used to identify a Diameter node uniquely. Before setting up diameter configuration, Netscaler (as a Diameter node) MUST be assigned a unique DiameterIdentity.

example =>

set ns diameter -identity netscaler.com

Now whenever Netscaler system needs to use identity in diameter messages. It will use 'netscaler.com' as Origin-Host AVP as defined in RFC3588

realm

Diameter Realm to be used by NS.

example =>

set ns diameter -realm com

Now whenever Netscaler system needs to use realm in diameter messages. It will use 'com' as Origin-Realm AVP as defined in RFC3588

serverClosePropagation

when a Server connection goes down, whether to close the corresponding client connection if there were requests pending on the server.

Possible values: YES, NO

Default value: NO

Use this command to remove ns diameter settings.Refer to the set ns diameter command for meanings of the arguments.

unset ns diameter -serverClosePropagation

Displays the diameter parameters configured on the NetScaler appliance.

show ns diameter

format

level

identity

DiameterIdentity to be used by NS. DiameterIdentity is used to identify a Diameter node uniquely. Before setting up diameter configuration, Netscaler (as a Diameter node) MUST be assigned a unique DiameterIdentity.

example =>

set ns diameter -identity netscaler.com

Now whenever Netscaler system needs to use identity in diameter messages. It will use 'netscaler.com' as Origin-Host AVP as defined in RFC3588

realm

Diameter Realm to be used by NS.

example =>

set ns diameter -realm com

Now whenever Netscaler system needs to use realm in diameter messages. It will use 'com' as Origin-Realm AVP as defined in RFC3588

serverClosePropagation

when a Server connection goes down, whether to close the corresponding client connection if there were requests pending on the server.

---

ns encryptionParams

Sep 22, 2015

The following operations can be performed on "ns encryptionParams":

set | show

Sets the parameters required for encrypting or decrypting content.

set ns encryptionParams -method <method> [-keyValue]

method

Cipher method (and key length) to be used to encrypt and decrypt content. The default value is AES256.

Possible values: NONE, RC4, DES3, AES128, AES192, AES256

keyValue

The base64-encoded key generation number, method, and key value.

Note:

* Do not include this argument if you are changing the encryption method.

* To generate a new key value for the current encryption method, specify an empty string \\(""\\) as the value of this parameter. The parameter is passed implicitly, with its automatically generated value, to the NetScaler packet engines even when it is not included in the command. Passing the parameter to the packet engines enables the appliance to save the key value to the configuration file and to propagate the key value to the secondary appliance in a high availability setup.

set ns encryptionParams -method aes128

Displays the encryption method configured on the NetScaler appliance.

show ns encryptionParams

format

level

method

The cipher method (and key length) used to encrypt and decrypt content.

keyValue

The base64-encoded key generation number, method, and key value.

Note:

* Do not include this argument if you are changing the encryption method.

* To generate a new key value for the current encryption method, specify an empty string \\(""\\) as the value of this parameter. The parameter is passed implicitly, with its automatically generated value, to the NetScaler packet engines even when it is not included in the command. Passing the parameter to the packet engines enables the appliance to save the key value to the configuration file and to propagate the key value to the secondary appliance in a high availability setup.

ns events

Sep 22, 2015

The following operations can be performed on "ns events":

Displays events that occur on the appliance.

show ns events [<eventNo>]

eventNo

Event number starting from which events must be shown.

time

Event no.

eventcode

event Code.

devid

Device Name.

devname

Device Name.

text

Event no.

data0

additional event information.

data1

additional event information.

data2

additional event information.

data3

additional event information.

devno

count

stateflag

show ns events

ns feature

Sep 22, 2015

The following operations can be performed on "ns feature":

enable | disable | show

enable ns feature

Enables NetScaler feature(s).

Synopsys

enable ns feature <feature> ...

Arguments

feature

Feature to be enabled. Multiple features can be specified by providing a blank space between each feature.

Example

enable ns feature sc This CLI command enables the SureConnect feature.

---

disable ns feature

Disables NetScaler feature(s).

Synopsys

disable ns feature <feature> ...

Arguments

feature

Feature to be disabled. Multiple features can be specified by providing a blank space between each feature.

---

show ns feature

Displays the current state of NetScaler features.

Synopsys

show ns feature

Outputs

feature

Feature to be enabled. Multiple features can be specified by providing a blank space between each feature.

WL

Web Logging.

SP

Surge Protection.

LB

Load Balancing.

CS

Content Switching.

CR

Cache Redirect.

SC

Sure Connect.

CMP

Compression.

PQ

Priority Queuing.

SSL

Secure Sockets Layer.

GSLB

Global Server Load Balancing.

HDOSP

DOS Protection.

Routing

Routing.NOTE: This attribute is deprecated.

CF

Content Filter.

IC

Integrated Caching.

SSLVPN

SSL VPN.

AAA

AAA

OSPF

OSPF Routing.

RIP

RIP Routing.

BGP

BGP Routing.

REWRITE

Rewrite.

IPv6PT

IPv6 protocol translation

AppFw

Application Firewall.

RESPONDER

Responder.

HTMLInjection

HTML Injection.

push

NetScaler Push.

AppFlow

AppFlow.

CloudBridge

CloudBridge.

ISIS

ISIS Routing.

CH

Call Home.

AppQoE

AppQoS

vPath

Vpath

ns hardware

Sep 22, 2015

The following operations can be performed on "ns hardware":

show ns hardware

Displays details of the appliance hardware and information such as the host ID and the serial number.

Synopsys

show ns hardware

Outputs

hwdescription

Hardware and it's ports detail.

sysId

System id.

manufactureDay

Manufacturing day.

manufactureMonth

Manufacturing month.

manufactureYear

Manufacturing year.

cpufrequncy

CPU Frequency.

hostId

host id.

host

host id.

serialNo

Serial no.

encodedSerialNo

Encoded serial no.

ns hostName

Sep 22, 2015

The following operations can be performed on "ns hostName":

set | show

set ns hostName

Sets the hostname for the NetScaler appliance. The hostname is displayed on the shell prompt.

Synopsys

set ns hostName <hostName> [-ownerNode <positive_integer>]

Arguments

hostName

Host name for the NetScaler appliance.

ownerNode

ID of the cluster node for which you are setting the hostname. Can be configured only through the cluster IP address.

Default value: 255

Maximum value: 31

Example

set ns hostname nspri

---

show ns hostName

Displays the host name of the system.

Synopsys

show ns hostName

Arguments

format

level

Outputs

hostName

Host name

ownerNode

ID of the cluster node for which you are setting the hostname. Can be configured only through the cluster IP address.

devno

count

stateflag

Example

show ns hostname

ns httpParam

Sep 22, 2015

The following operations can be performed on "ns httpParam":

set | unset | show

set ns httpParam

Sets the configurable HTTP parameters for the NetScaler appliance.

Synopsys

set ns httpParam [-dropInvalReqs (ON | OFF)] [-markHttp09Inval (ON | OFF)] [-markConnReqInval (ON | OFF)] [-insNsSrvrHdr (ON | OFF) [<nsSrvrHdr>]] [-logErrResp (ON | OFF)] [-conMultiplex (ENABLED | DISABLED)] [-maxReusePool <positive_integer>]

Arguments

dropInvalReqs

Drop invalid HTTP requests or responses.

Possible values: ON, OFF

Default value: OFF

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

Possible values: ON, OFF

Default value: OFF

markConnReqInval

Mark CONNECT requests as invalid.

Possible values: ON, OFF

Default value: OFF

insNsSrvrHdr

Enable or disable NetScaler server header insertion for NetScaler generated HTTP responses.

Possible values: ON, OFF

Default value: OFF

logErrResp

Server header value to be inserted.

Possible values: ON, OFF

Default value: ON

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

Maximum value: 360000

Example

```
set ns httpParam -dropInvalReqs ON
```

---

unset ns httpParam

Use this command to remove ns httpParam settings.Refer to the set ns httpParam command for meanings of the arguments.

Synopsys

```
unset ns httpParam [-dropInvalReqs] [-markHttp09Inval] [-markConnReqInval] [-insNsSrvrHdr] [-nsSrvrHdr] [-logErrResp] [-conMultiplex] [-maxReusePool]
```

---

show ns httpParam

Displays the HTTP parameters configured on the NetScaler appliance.

Synopsys

```
show ns httpParam
```

Arguments

format

level

Outputs

dropInvalReqs

Drop invalid HTTP requests or responses.

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

markConnReqInval

Mark CONNECT requests as invalid.

insNsSrvrHdr

Enable or disable NetScaler server header insertion for NetScaler generated HTTP responses.

nsSrvrHdr

The server header value to be inserted. If no explicit header is specified then NSBUILD.RELEASE is used as default server header.

logErrResp

Whether to log HTTP error responses

conMultiplex

Reuse server connections for requests from more than one client connections.

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

ns httpProfile

Sep 22, 2015

The following operations can be performed on "ns httpProfile":

add | rm | set | unset | show

add ns httpProfile

Adds an HTTP profile to the NetScaler appliance.

Synopsys

add ns httpProfile <name> [-dropInvalReqs (ENABLED | DISABLED)] [-markHttp09Inval (ENABLED | DISABLED)] [-markConnReqInval (ENABLED | DISABLED)] [-cmpOnPush (ENABLED | DISABLED)] [-conMultiplex (ENABLED | DISABLED)] [-maxReusePool <positive_integer>] [-dropExtraCRLF (ENABLED | DISABLED)] [-incompHdrDelay <positive_integer>] [-webSocket (ENABLED | DISABLED)] [-rtspTunnel (ENABLED | DISABLED)] [-reqTimeout <positive_integer>] [-adptTimeout (ENABLED | DISABLED)] [-reqTimeoutAction <string>] [-dropExtraData (ENABLED | DISABLED)] [-webLog (ENABLED | DISABLED)] [-clientIpHdrExpr <expression>] [-maxReq <positive_integer>] [-persistentETag (ENABLED | DISABLED)] [-spdy (ENABLED | DISABLED)] [-reusePoolTimeout <positive_integer>] [-maxHeaderLen <positive_integer>]

Arguments

name

Name for an HTTP profile. Must begin with a letter, number, or the underscore \\(_\\) character. Other characters allowed, after the first character, are the hyphen \\(-\\), period \\(.\\), hash \\(\\#\\), space \\(\\), at \\(@\\), and equal \\(=\\) characters. The name of a HTTP profile cannot be changed after it is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my http profile" or 'my http profile'\\).

dropInvalReqs

Drop invalid HTTP requests or responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markConnReqInval

Mark CONNECT requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cmpOnPush

Start data compression on receiving a TCP packet with PUSH flag set.

Possible values: ENABLED, DISABLED

Default value: DISABLED

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

Maximum value: 360000

dropExtraCRLF

Drop any extra 'CR' and 'LF' characters present after the header.

Possible values: ENABLED, DISABLED

Default value: ENABLED

incompHdrDelay

Maximum time to wait, in milliseconds, between incomplete header packets. If the header packets take longer to arrive at NetScaler, the connection is silently dropped.

Default value: 7000

Maximum value: 360000

webSocket

HTTP connection to be upgraded to a web socket connection. Once upgraded, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rtspTunnel

Allow RTSP tunnel in HTTP. Once application/x-rtsp-tunnelled is seen in Accept or Content-Type header, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeout

Time, in seconds, within which the HTTP request must complete. If the request does not complete within this time, the specified request timeout action is executed.

Maximum value: 86400

adptTimeout

Adapts the configured request timeout based on flow conditions. The timeout is increased or decreased internally and applied on the flow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeoutAction

Action to take when the HTTP request does not complete within the specified request timeout duration. You can configure the following actions:

* RESET - Send RST (reset) to client when timeout occurs.

* DROP - Drop silently when timeout occurs.

* Custom responder action - Name of the responder action to trigger when timeout occurs, used to send custom message.

dropExtraData

Drop any extra data when server sends more data than the specified content-length.

Possible values: ENABLED, DISABLED

Default value: DISABLED

webLog

Enable or disable web logging.

Possible values: ENABLED, DISABLED

Default value: ENABLED

clientIpHdrExpr

Name of the header that contains the real client IP address.

maxReq

Maximum requests allowed on a single connection.

Maximum value: 65534

persistentETag

Generate the persistent NetScaler specific ETag for the HTTP response with ETag header.

Possible values: ENABLED, DISABLED

Default value: DISABLED

spdy

Allow SPDY over SSL vserver. SSL will advertise SPDY support during NPN Handshake.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reusePoolTimeout

Idle timeout (in seconds) for server connections in re-use pool. Connections in the re-use pool are flushed, if they remain idle for the configured timeout.

Maximum value: 31536000

maxHeaderLen

Number of bytes to be queued to look for complete header before returning error. If complete header is not obtained after queuing these many bytes, request will be marked as invalid and no L7 processing will be done for that TCP connection.

Default value: 24820

Minimum value: 2048

Maximum value: 61440

Example

```
add httpprofile <profile name> -dropInvalReqs ON -markHttp09Inval ON
```

---

rm ns httpProfile

Removes an HTTP profile from the appliance.

Synopsys

```
rm ns httpProfile <name>
```

Arguments

name

Name of the HTTP profile to be removed.

Example

rm httpprofile <profile name>

---

set ns httpProfile

Modifies the attributes of an HTTP profile.

Synopsys

set ns httpProfile <name> [-dropInvalReqs (ENABLED | DISABLED)] [-markHttp09Inval (ENABLED | DISABLED)] [-markConnReqInval (ENABLED | DISABLED)] [-cmpOnPush (ENABLED | DISABLED)] [-conMultiplex (ENABLED | DISABLED)] [-maxReusePool <positive_integer>] [-dropExtraCRLF (ENABLED | DISABLED)] [-incompHdrDelay <positive_integer>] [-webSocket (ENABLED | DISABLED)] [-rtspTunnel (ENABLED | DISABLED)] [-reqTimeout <positive_integer>] [-adptTimeout (ENABLED | DISABLED)] [-reqTimeoutAction <string>] [-dropExtraData (ENABLED | DISABLED)] [-webLog (ENABLED | DISABLED)] [-clientIpHdrExpr <expression>] [-maxReq <positive_integer>] [-persistentETag (ENABLED | DISABLED)] [-spdy (ENABLED | DISABLED)] [-reusePoolTimeout <positive_integer>] [-maxHeaderLen <positive_integer>]

Arguments

name

Name of the HTTP profile to be modified.

dropInvalReqs

Drop invalid HTTP requests or responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markConnReqInval

Mark CONNECT requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cmpOnPush

Start data compression on receiving a TCP packet with PUSH flag set.

Possible values: ENABLED, DISABLED

Default value: DISABLED

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

Maximum value: 360000

dropExtraCRLF

Drop any extra 'CR' and 'LF' characters present after the header.

Possible values: ENABLED, DISABLED

Default value: ENABLED

incompHdrDelay

Maximum time to wait, in milliseconds, between incomplete header packets. If the header packets take longer to arrive at NetScaler, the connection is silently dropped.

Default value: 7000

Maximum value: 360000

webSocket

HTTP connection to be upgraded to a web socket connection. Once upgraded, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rtspTunnel

Allow RTSP tunnel in HTTP. Once application/x-rtsp-tunnelled is seen in Accept or Content-Type header, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeout

Time, in seconds, within which the HTTP request must complete. If the request does not complete within this time, the specified request timeout action is executed.

Maximum value: 86400

adptTimeout

Adapts the configured request timeout based on flow conditions. The timeout is increased or decreased internally and applied on the flow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeoutAction

Action to take when the HTTP request does not complete within the specified request timeout duration. You can configure the following actions:

* RESET - Send RST (reset) to client when timeout occurs.

* DROP - Drop silently when timeout occurs.

* Custom responder action - Name of the responder action to trigger when timeout occurs, used to send custom message.

dropExtraData

Drop any extra data when server sends more data than the specified content-length.

Possible values: ENABLED, DISABLED

Default value: DISABLED

webLog

Enable or disable web logging.

Possible values: ENABLED, DISABLED

Default value: ENABLED

clientIpHdrExpr

Name of the header that contains the real client IP address.

maxReq

Maximum requests allowed on a single connection.

Maximum value: 65534

persistentETag

Generate the persistent NetScaler specific ETag for the HTTP response with ETag header.

Possible values: ENABLED, DISABLED

Default value: DISABLED

spdy

Allow SPDY over SSL vserver. SSL will advertise SPDY support during NPN Handshake.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reusePoolTimeout

Idle timeout (in seconds) for server connections in re-use pool. Connections in the re-use pool are flushed, if they remain idle for the configured timeout.

Maximum value: 31536000

maxHeaderLen

Number of bytes to be queued to look for complete header before returning error. If complete header is not obtained after queuing these many bytes, request will be marked as invalid and no L7 processing will be done for that TCP connection.

Default value: 24820

Minimum value: 2048

Maximum value: 61440

Example

```
set httpprofile <profile name> -dropInvalReqs ON -markHttp09Inval ON
```

---

unset ns httpProfile

Removes the attributes of the HTTP profile. Attributes for which a default value is available revert to their default values. Refer to the 'set ns httpProfile' command for a description of the parameters..Refer to the set ns httpProfile command for meanings of the arguments.

Synopsys

```
unset ns httpProfile <name> [-dropInvalReqs] [-markHttp09Inval] [-markConnReqInval] [-cmpOnPush] [-conMultiplex] [-maxReusePool] [-dropExtraCRLF] [-incompHdrDelay] [-webSocket] [-dropExtraData] [-clientIpHdrExpr] [-reqTimeout] [-adptTimeout] [-reqTimeoutAction] [-webLog] [-maxReq] [-persistentETag] [-spdy] [-reusePoolTimeout] [-maxHeaderLen] [-rtspTunnel]
```

show ns httpProfile

Displays information about HTTP profiles configured on the appliance.

Synopsys

show ns httpProfile [<name>]

Arguments

name

Name of the HTTP profile to be displayed. If a name is not provided, information about all HTTP profiles is shown.

summary

fullValues

format

level

Outputs

dropInvalReqs

Dropping of invalid HTTP requests/responses

markHttp09Inval

Invalidating HTTP 0.9 requests

markConnReqInval

Invalidating CONNECT HTTP requests

cmpOnPush

Compression on PUSH packet

conMultiplex

Reuse server connections for requests from more than one client connections.

maxReusePool

Maximum connections in reusepool

webSocket

HTTP connection to be upgraded to a web socket connection. Once upgraded, NetScaler does not process Layer 7 traffic on this connection.

refCnt

Number of entities using this profile

stateflag

State flag

dropExtraCRLF

Drop any extra 'CR' and 'LF' characters present after the header.

incompHdrDelay

Maximum time to wait, in milliseconds, between incomplete header packets. If the header packets take longer to arrive at NetScaler, the connection is silently dropped.

reqTimeout

Time, in seconds, within which the HTTP request must complete. If the request does not complete within this time, the specified request timeout action is executed.

adptTimeout

Adapts the configured request timeout based on flow conditions. The timeout is increased or decreased internally and applied on the flow.

reqTimeoutAction

Action to take when the HTTP request does not complete within the specified request timeout duration. You can configure the following actions:

* RESET - Send RST (reset) to client when timeout occurs.

* DROP - Drop silently when timeout occurs.

* Custom responder action - Name of the responder action to trigger when timeout occurs, used to send custom message.

dropExtraData

Drop any extra data when server sends more data than the specified content-length.

webLog

Disabling weblog option

clientIpHdrExpr

Name of the header that contains the real client IP address.

maxReq

Maximum requests allowed on a single connection.

persistentETag

Generate the persistent NetScaler specific ETag for the HTTP response with ETag header.

spdy

Allow SPDY over SSL vserver. SSL will advertise SPDY support during NPN Handshake.

reusePoolTimeout

Idle timeout (in seconds) for server connections in re-use pool. Connections in the re-use pool are flushed, if they remain idle for the configured timeout.

maxHeaderLen

Number of bytes to be queued to look for complete header before returning error. If complete header is not obtained after queuing these many bytes, request will be marked as invalid and no L7 processing will be done for that TCP connection.

rtspTunnel

Allow RTSP tunnel in HTTP. Once application/x-rtsp-tunnelled is seen in Accept or Content-Type header, NetScaler does not process Layer 7 traffic on this connection.

devno

count

Example

```
show http profile [profile name]
```

ns idletimeout

Sep 22, 2015

The following operations can be performed on "ns idletimeout":

set | unset | show

set ns idletimeout

Set the pcb/natpcb idletimeout. NOTE: This command is deprecated.

Synopsys

Arguments

tcpsvr

Set the idletimeout for server side pcb.

tcpclt

Set the idletimeout for client side pcb.

nontcpsvrclt

Set the idletimeout for natpcb.

Default value: 120

Minimum value: 1

Example

```
set ns idletimeout -tcpsvr 120  set ns idletimeout -tcpclt 120  set ns idletimeout -nontcpsvrclt 120
```

---

unset ns idletimeout

Use this command to remove ns idletimeout settings.Refer to the set ns idletimeout command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

---

show ns idletimeout

Display the global setting of pcb/natpcb idletimeout. NOTE: This command is deprecated.This command is deprecated in favour of 'set ns timeout'

Synopsys

Arguments

format

level

Outputs

tcpsvr

Set the idletimeout for server side pcb.

tcpclt

Set the idletimeout for client side pcb.

nontcpsvrclt

Set the idletimeout for natpcb.

---

ns info

Sep 22, 2015

The following operations can be performed on "ns info":

show ns info

Displays the following details of the NetScaler appliance: * Software version * NetScaler IP address and subnet mask * Number of mapped IP addresses * Identifies the appliance as a standalone appliance, a part of an HA pair, or is a cluster node * Current time on the system and timestamp when the appliance was last updated * Features that are enabled or disabled * Modes that are enabled or disabled

Synopsys

show ns info

Example

An example of this command's output is shown below: System Rainier: Build 24, Date: Apr 25 2002, 21:13:25 System IP: 10.101.4.22 (mask: 255.255.0.0) Mapped IP: 10.1

ns ip

Sep 22, 2015

The following operations can be performed on "ns ip":

add | rm | set | unset | enable | disable | show

add ns ip

Creates an IPv4 address on the NetScaler appliance.

Synopsys

add ns ip <IPAddress>@ <netmask> [-type <type> [-hostRoute (ENABLED | DISABLED) [-hostRtGw <ip_addr>] [-metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-ospfLSAType (TYPE1 | TYPE5) [-ospfArea <positive_integer>]]]] [-arp (ENABLED | DISABLED)] [-icmp (ENABLED | DISABLED)] [-vServer (ENABLED | DISABLED)] [-telnet (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-gui <gui>] [-ssh (ENABLED | DISABLED)] [-snmp (ENABLED | DISABLED)] [-mgmtAccess (ENABLED | DISABLED)] [-restrictAccess (ENABLED | DISABLED)] [-dynamicRouting (ENABLED | DISABLED)] [-state (ENABLED | DISABLED)] [-vrID <positive_integer>] [-icmpResponse <icmpResponse>] [-ownerNode <positive_integer>] [-arpResponse <arpResponse>] [-td <positive_integer>]

Arguments

IPAddress

IPv4 address to create on the NetScaler appliance. Cannot be changed after the IP address is created.

netmask

Subnet mask associated with the IP address.

type

Type of the IP address to create on the NetScaler appliance. Cannot be changed after the IP address is created.

Possible values: SNIP, VIP, MIP, NSIP, GSLBsiteIP, CLIP

Default value: NSADDR_SNIP

arp

Respond to respond to ARP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

Respond to ICMP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

Use this option to set (enable or disable) the vserver attribute for this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

Allow Telnet access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

Allow File Transfer Protocol (FTP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

Allow graphical user interface (GUI) access to this IP address.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

Allow secure shell (SSH) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

Allow Simple Network Management Protocol (SNMP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

Allow access to management applications on this IP address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Block access to nonmanagement applications on this IP. This option is applicable for MIPs, SNIPs, and NSIP, and is disabled by default. Nonmanagement applications can run on the underlying NetScaler Free BSD operating system.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicRouting

Allow dynamic routing on the IP address. Specific to SNIP addresses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ospf

Use this option to enable or disable OSPF on this IP address for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

bgp

Use this option to enable or disable BGP on this IP address for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rip

Use this option to enable or disable RIP on this IP address for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP address, using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

hostRtGw

IP address of the gateway for the route. If hostRtGw is not set(i.e. default value), VIP goes with MIP as the gateway(if MIP is present) or with 0.0.0.0 (if MIP is not present).

Default value: -1

metric

Integer value to add to or subtract from the cost of the route advertised for the VIP address.

Minimum value: -16777215

vserverRHILevel

Advertise the route for the Virtual IP (VIP) address on the basis of the state of the virtual servers associated with that VIP.

* NONE - Advertise the route for the VIP address, regardless of the state of the virtual servers associated with the address.

* ONE VSERVER - Advertise the route for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - Advertise the route for the VIP address if all of the associated virtual servers are in UP state.

Possible values: ONE_VSERVER, ALL_VSERVERS, NONE

Default value: RHI_STATE_ONE

ospfLSAType

Type of LSAs to be used by the OSPF protocol, running on the NetScaler appliance, for advertising the route for this VIP address.

Possible values: TYPE1, TYPE5

Default value: DISABLED

ospfArea

ID of the area in which the Type1 LSAs are to be advertised for this VIP address by the OSPF protocol running on the NetScaler appliance. When ospfArea is not set, the VIP is advertised on all areas.

Default value: -1

Maximum value: 4294967294LU

state

Enable or disable the IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vrID

ID, which uniquely identifies a VMAC address, to bin to a VIP address. This binding is used to set up NetScaler appliances in an active-active configuration using VRRP.

Minimum value: 1

Maximum value: 255

icmpResponse

Respond to ICMP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ICMP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if all of the associated virtual servers are in UP state.

* VSVR_CNTRLD - The behavior depends on the ICMP VSERVER RESPONSE setting on all the associated virtual servers.

The following settings can be made for the ICMP VSERVER RESPONSE parameter on a virtual server:

* If you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.

* If you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds if even one virtual server is UP.

* When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds if even one virtual server set to ACTIVE is UP.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS, VSVR_CNTRLD

Default value: NS_IP_NONE

ownerNode

The owner node in a Cluster for this IP address. Owner node can vary from 0 to 31. If ownernode is not specified then the IP is treated as Striped IP.

Default value: 255

arpResponse

Respond to ARP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS

Default value: NS_IP_NONE

td

Traffic Domain Id

Maximum value: 4094

Example

```
add ns ip 10.102.4.123 255.255.255.0
```

rm ns ip

Removes an IPv4 address configured on the NetScaler appliance.

Synopsys

rm ns ip <IPAddress>@ [-td <positive_integer>]

Arguments

IPAddress

IPv4 address that you want to remove.

td

Traffic Domain Id

Maximum value: 4094

Example

rm ns ip 10.102.4.123

---

set ns ip

Modifies the parameters of an IPv4 address configured on the NetScaler appliance.

Synopsys

set ns ip (<IPAddress>@ [-td <positive_integer>]) [-netmask <netmask>] [-arp (ENABLED | DISABLED)] [-icmp (ENABLED | DISABLED)] [-vServer (ENABLED | DISABLED)] [-telnet (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-gui <gui>] [-ssh (ENABLED | DISABLED)] [-snmp (ENABLED | DISABLED)] [-mgmtAccess (ENABLED | DISABLED)] [-restrictAccess (ENABLED | DISABLED)] [-dynamicRouting (ENABLED | DISABLED)] [-hostRoute (ENABLED | DISABLED) [-hostRtGw <ip_addr>] [-metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-ospfLSAType (TYPE1 | TYPE5) [-ospfArea <positive_integer>]]] [-vrID <positive_integer>] [-icmpResponse <icmpResponse>] [-arpResponse <arpResponse>]

Arguments

IPAddress

IPv4 address whose parameters you want to modify.

netmask

Subnet mask associated with the IP address.

arp

Respond to respond to ARP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

Respond to ICMP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

Use this option to set (enable or disable) the vserver attribute for this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

Allow Telnet access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

Allow File Transfer Protocol (FTP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

Allow graphical user interface (GUI) access to this IP address.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

Allow secure shell (SSH) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

Allow Simple Network Management Protocol (SNMP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

Allow access to management applications on this IP address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Block access to nonmanagement applications on this IP. This option is applicable for MIPs, SNIPs, and NSIP, and is disabled by default. Nonmanagement applications can run on the underlying NetScaler Free BSD operating system.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicRouting

Allow dynamic routing on the IP address. Specific to SNIP addresses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ospf

The state of OSPF on this IP address for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

bgp

The state of BGP on this IP address for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rip

The state of RIP on this IP address for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP address, using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

vrID

ID, which uniquely identifies a VMAC address, to bin to a VIP address. This binding is used to set up NetScaler appliances in an active-active configuration using VRRP.

Minimum value: 1

Maximum value: 255

icmpResponse

Respond to ICMP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ICMP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if all of the associated virtual servers are in UP state.

* VSVR_CNTRLD - The behavior depends on the ICMP VSERVER RESPONSE setting on all the associated virtual servers.

The following settings can be made for the ICMP VSERVER RESPONSE parameter on a virtual server:

* If you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.

* If you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds if even one virtual server is UP.

* When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds if even one virtual server set to ACTIVE is UP.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS, VSVR_CNTRLD

Default value: NS_IP_NONE

arpResponse

Respond to ARP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS

Default value: NS_IP_NONE

Example

```
set ns ip 10.102.4.123 -arp ENABLED
```

---

unset ns ip

Modifies the parameters of an IPv4 address configured on the NetScaler appliance..Refer to the set ns ip command for meanings of the arguments.

Synopsys

```
unset ns ip <IPAddress>@ [-td <positive_integer>] [-ospfArea] [-hostRtGw] [-netmask] [-arp] [-icmp] [-vServer] [-telnet] [-ftp] [-gui] [-ssh] [-snmp] [-mgmtAccess] [-restrictAccess] [-dynamicRouting] [-hostRoute] [-metric] [-vserverRHILevel] [-ospfLSAType] [-vrID] [-icmpResponse] [-arpResponse]
```

Example

```
unset ns ip 10.102.4.123 -ospfArea
```

---

enable ns ip

Enables the specified IP address configured on the NetScaler appliance.

Synopsys

```
enable ns ip (<IPAddress>@ [-td <positive_integer>])
```

Arguments

IPAddress

IP address that you want to enable.

Example

```
enable ns ip 10.10.10.10
```

---

disable ns ip

Disables the specified IP address configured on the NetScaler appliance.

Synopsys

```
disable ns ip (<IPAddress>@ [-td <positive_integer>])
```

Arguments

IPAddress

IP address that you want to disable.

Example

```
disable ns ip 10.10.10.10
```

---

show ns ip

Displays settings of all the IPv4 addresses or of the specified IPv4 address configured on the NetScaler appliance. To display settings of all the IPv4 addresses, run the command without any parameters. To display settings of a particular IPv4 address, specify the IPv4 address.

Synopsys

show ns ip [<IPAddress> [-td <positive_integer>]] [-type <type>]

Arguments

IPAddress

IPv4 address whose details you want the NetScaler appliance to display.

type

Display the settings of all IPv4 addresses of a particular type.

Possible values: SNIP, VIP, MIP, NSIP, GSLBsiteIP, CLIP

summary

fullValues

format

level

Outputs

netmask

The netmask of this IP.

flags

The flags for this entry.

ipAttribute

arp

Whether arp is enabled or disabled.

icmp

Whether icmp is enabled or disabled.

vServer

Whether vserver is enabled or disabled.

telnet

Whether telnet is enabled or disabled.

ssh

Whether ssh is enabled or disabled.

gui

Whether gui is (enabled | SecureOnly | disabled).

snmp

Whether snmp is enabled or disabled.

ftp

Whether ftp is enabled or disabled.

mgmtAccess

Whether management access is enabled or disabled.

restrictAccess

Blocking of all ports not used for management access enabled or disabled

dynamicRouting

Whether dynamic routing is enabled or disabled.

bgp

Whether bgp is enabled or disabled.NOTE: This attribute is deprecated.

ospf

Whether ospf is enabled or disabled.NOTE: This attribute is deprecated.

rip

Whether rip is enabled or disabled.NOTE: This attribute is deprecated.

hostRoute

Whether host route is enabled or disabled.

hostRtGw

Gateway used for advertising host route.

hostRtGwAct

Actual Gateway used for advertising host route.

metric

The metric value added or subtracted from the cost of the hostroute.

ospfArea

The area ID of the area in which OSPF Type1 LSAs are advertised. When ospfArea if not set, LSAs are advertised on all areas.NOTE: This attribute is deprecated.Replaced by ospfAreaval

ospfAreaval

The area ID of the area in which OSPF Type1 LSAs are advertised.

vserverRHILevel

The rhi level for this IP.

VIPrtadv2BSD

Whether this route is advertised to FreeBSD

VIPvserCount

Number of vservers bound to this VIP

VIPvserDownCount

Number of vservers bound to this VIP, which are down

ospfLSAType

The ospf lsa type to use while advertising this IP.

state

Whether this ip is enabled or disabled.

freePorts

Number of free Ports available on this IP

vrID

ID, which uniquely identifies a VMAC address, to bin to a VIP address. This binding is used to set up NetScaler appliances in an active-active configuration using VRRP.

ipType

icmpResponse

Respond to ICMP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ICMP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if all of the associated virtual servers are in UP state.

* VSVR_CNTRLD - The behavior depends on the ICMP VSERVER RESPONSE setting on all the associated virtual servers.

The following settings can be made for the ICMP VSERVER RESPONSE parameter on a virtual server:

* If you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.

* If you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds if even one virtual server is UP.

* When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds if even one virtual server set to ACTIVE is UP.

ownerNode

The owner node in a Cluster for this IP address. Owner node can vary from 0 to 31. If ownernode is not specified then the IP is treated as Striped IP.

arpResponse

Respond to ARP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

stateflag

cfgflags

This contains the flags for IP in DB

ipRefcount

Used to keep reference count of IP

devno

count

Example

show ns ip Ipaddress Type Mode Arp Icmp Vserver State Owner --------- ---- ---- --- ---- ------- ----- ----- 1)10.102.169

ns ip6

Sep 22, 2015

The following operations can be performed on "ns ip6":

add | rm | set | unset | show

add ns ip6

Creates an IPv6 address on the NetScaler appliance.

Synopsys

add ns ip6 <IPv6Address>@ [-scope (global | link-local)] [-type <type> [-hostRoute (ENABLED | DISABLED) [-ip6hostRtGw <ipv6_addr|*>] [-metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-ospf6LSAType (INTRA_AREA | EXTERNAL) [-ospfArea <positive_integer>]]]] [-vlan <positive_integer>] [-nd (ENABLED | DISABLED)] [-icmp (ENABLED | DISABLED)] [-vServer (ENABLED | DISABLED)] [-telnet (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-gui <gui>] [-ssh (ENABLED | DISABLED)] [-snmp (ENABLED | DISABLED)] [-mgmtAccess (ENABLED | DISABLED)] [-restrictAccess (ENABLED | DISABLED)] [-dynamicRouting (ENABLED | DISABLED)] [-state (DISABLED | ENABLED)] [-map <ip_addr>] [-ownerNode <positive_integer>] [-td <positive_integer>]

Arguments

IPv6Address

IPv6 address to create on the NetScaler appliance.

scope

Scope of the IPv6 address to be created. Cannot be changed after the IP address is created.

Possible values: global, link-local

Default value: NS_GLOBAL

type

Type of IP address to be created on the NetScaler appliance. Cannot be changed after the IP address is created.

Possible values: NSIP, VIP, SNIP, GSLBsiteIP, ADNSsvcIP, CLIP

Default value: NS_IPV6_SNIP

vlan

The VLAN number.

Maximum value: 4094

nd

Respond to Neighbor Discovery (ND) requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

Respond to ICMP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

Enable or disable the state of all the virtual servers associated with this VIP6 address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

Allow Telnet access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

Allow File Transfer Protocol (FTP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

Allow graphical user interface (GUI) access to this IP address.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

Allow secure Shell (SSH) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

Allow Simple Network Management Protocol (SNMP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

Allow access to management applications on this IP address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Block access to nonmanagement applications on this IP address. This option is applicable forMIP6s, SNIP6s, and NSIP6s, and is disabled by default. Nonmanagement applications can run on the underlying NetScaler Free BSD operating system.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicRouting

Allow dynamic routing on the IP address. Specific to SNIP6 address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP6 address by using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

ip6hostRtGw

IPv6 address of the gateway for the route. If Gateway is not set, VIP uses :: as the gateway.

metric

Integer value to add to or subtract from the cost of the route advertised for the VIP6 address.

Minimum value: -16777215

vserverRHILevel

Advertise or do not advertise the route for the Virtual IP (VIP6) address on the basis of the state of the virtual servers associated with that VIP6.

* NONE - Advertise the route for the VIP6 address, irrespective of the state of the virtual servers associated with the address.

* ONE VSERVER - Advertise the route for the VIP6 address if at least one of the associated virtual servers is in UP

state.

* ALL VSERVER - Advertise the route for the VIP6 address if all of the associated virtual servers are in UP state.

Possible values: ONE_VSERVER, ALL_VSERVERS, NONE

Default value: RHI_STATE_ONE

ospf6LSAType

Type of LSAs to be used by the IPv6 OSPF protocol, running on the NetScaler appliance, for advertising the route for the VIP6 address.

Possible values: INTRA_AREA, EXTERNAL

Default value: DISABLED

ospfArea

ID of the area in which the Intra-Area-Prefix LSAs are to be advertised for the VIP6 address by the IPv6 OSPF protocol running on the NetScaler appliance. When ospfArea is not set, VIP6 is advertised on all areas.

Default value: -1

Maximum value: 4294967294LU

state

Enable or disable the IP address.

Possible values: DISABLED, ENABLED

Default value: ENABLED

map

Mapped IPV4 address for the IPV6 address.

ownerNode

ID of the cluster node for which you are adding the IP address. Must be used if you want the IP address to be active only on the specific node. Can be configured only through the cluster IP address. Cannot be changed after the IP address is created.

Default value: 255

td

Traffic Domain Id

Maximum value: 4094

Example

```
add ns ip6 2001::a/96 -scope GLOBAL
```

rm ns ip6

Removes an IPv6 address configured on the NetScaler appliance.

Synopsys

rm ns ip6 <IPv6Address>@ [-td <positive_integer>]

Arguments

IPv6Address

IPv6 address that you want to remove.

td

Traffic Domain Id

Maximum value: 4094

Example

rm ns ip6 2002::5

set ns ip6

Modifies the specified parameters of an IPv6 address configured on the NetScaler appliance.

Synopsys

set ns ip6 (<IPv6Address>@ [-td <positive_integer>]) [-nd (ENABLED | DISABLED)] [-icmp (ENABLED | DISABLED)] [-vServer (ENABLED | DISABLED)] [-telnet (ENABLED | DISABLED)] [-ftp (ENABLED | DISABLED)] [-gui <gui>] [-ssh (ENABLED | DISABLED)] [-snmp (ENABLED | DISABLED)] [-mgmtAccess (ENABLED | DISABLED)] [-restrictAccess (ENABLED | DISABLED)] [-state (DISABLED | ENABLED)] [-map <ip_addr>] [-dynamicRouting (ENABLED | DISABLED)] [-hostRoute (ENABLED | DISABLED) [-ip6hostRtGw <ipv6_addr|*>] [-metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-ospf6LSAType (INTRA_AREA | EXTERNAL) [-ospfArea <positive_integer>]]]

Arguments

IPv6Address

IPv6 address whose parameters you want to modify.

nd

The state of ND responses for the entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

The state of ICMP responses for the entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

The state of vserver attribute for this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

The state of telnet access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

The state of ftp access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

The state of GUI access to this IP entity.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

The state of SSH access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

The state of SNMP access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

The state of management access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Status of ports not used for management access (blocked/open) for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

state

Enable or disable the IP address.

Possible values: DISABLED, ENABLED

Default value: ENABLED

map

Mapped IPV4 address for the IPV6 address.

dynamicRouting

Allow dynamic routing on the IP address. Specific to SNIP6 address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP6 address by using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

Example

```
set ns ip6 2001::a -map 10.102.33.27
```

---

unset ns ip6

Modifies the parameters of an IPv6 address configured on the NetScaler appliance..Refer to the set ns ip6 command for meanings of the arguments.

Synopsys

unset ns ip6 <IPv6Address>@ [-td <positive_integer>] [-ospfArea] [-nd] [-icmp] [-vServer] [-telnet] [-ftp] [-gui] [-ssh] [-snmp] [-mgmtAccess] [-restrictAccess] [-state] [-map] [-dynamicRouting] [-hostRoute] [-ip6hostRtGw] [-metric] [-vserverRHILevel] [-ospf6LSAType]

Example

unset ns ip6 2001::a -ospfArea

---

show ns ip6

Displays settings of all the IPv6 addresses or of the specified IPv6 address configured on the NetScaler appliance. To display settings of all the IPv6 addresses, run the command without any parameters. To display settings of a particular IPv6 address, specify the IPv6 address.

Synopsys

show ns ip6 [<IPv6Address> [-td <positive_integer>]]

Arguments

IPv6Address

IPv6 address whose settings you want the NetScaler appliance to display.

summary

fullValues

format

level

Outputs

scope

Scope of the IPv6 address to be created. Cannot be changed after the IP address is created.

type

The type of the IPV6 addressNOTE: This attribute is deprecated.This option is deprecated in favour of ip6_type

ipType

The type of the IPv6 address

vlan

The VLAN number.

nd

Whether ND is enabled or disabled.

icmp

Whether icmp is enabled or disabled.

vServer

Whether vserver is enabled or disabled.

telnet

Whether telnet is enabled or disabled.

ssh

Whether ssh is enabled or disabled.

gui

Whether gui is (enabled | SecureOnly | disabled).

snmp

Whether snmp is enabled or disabled.

ftp

Whether ftp is enabled or disabled.

mgmtAccess

Whether management access is enabled or disabled.

restrictAccess

Blocking of all ports not used for management access enabled or disabled

state

Current state of this IP.

map

Mapped IPV4 address for the IPV6 address.

dynamicRouting

Allow dynamic routing on the IP address. Specific to SNIP6 address.

hostRoute

Advertise a route for the VIP6 address by using the dynamic routing protocols running on the NetScaler appliance.

ip6hostRtGw

IPv6 address of the gateway for the route. If Gateway is not set, VIP uses :: as the gateway.

metric

The metric value to be added or subtracted from the cost of the hostroute advertised for this IPv6 entity.

vserverRHILevel

Advertise or do not advertise the route for the Virtual IP (VIP6) address on the basis of the state of the virtual servers associated with that VIP6.

* NONE - Advertise the route for the VIP6 address, irrespective of the state of the virtual servers associated with the address.

* ONE VSERVER - Advertise the route for the VIP6 address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - Advertise the route for the VIP6 address if all of the associated virtual servers are in UP state.

VIPrtadv2BSD

Whether this route is advertised to FreeBSD

VIPvserCount

Number of vservers bound to this VIP

VIPvserDownCount

Number of vservers bound to this VIP, which are down

ospf6LSAType

The OSPF's route advertisement type.

ospfArea

The area ID of the area in which OSPF INTRA AREA PREFIX LSAs should be advertised. When ospfArea is not set, LSAs are advertised in all areas.

ownerNode

ID of the cluster node for which you are adding the IP address. Must be used if you want the IP address to be active only on the specific node. Can be configured only through the cluster IP address. Cannot be changed after the IP address is created.

stateflag

cfgflags

This contains the flags for IP in DB

ipRefcount

Used to keep reference count of IPv6

systemType

The type of the System. Possible Values: Standalone, HA, Cluster. Used for display purpose.

devno

count

Example

show ns ip6

ns license

Sep 22, 2015

The following operations can be performed on "ns license":

show ns license

Displays the state of all the licensed features.

Synopsys

show ns license

Outputs

WL

Web Logging.

SP

Surge Protection.

LB

Load Balancing.

CS

Content Switching.

CR

Cache Redirect.

SC

Sure Connect.

CMP

Compression.

DELTA

Delta Compression.

PQ

Priority Queuing.

SSL

Secure Sockets Layer.

GSLB

Global Server Load Balancing.

GSLBP

GSLB Proximity.

HDOSP

DOS Protection.

Routing

Routing.

CF

Content Filter.

IC

Integrated Caching.

SSLVPN

SSL VPN.

AAA

AAA

OSPF

OSPF Routing.

RIP

RIP Routing.

BGP

BGP Routing.

REWRITE

Rewrite.

IPv6PT

IPv6 protocol translation

AppFw

Application Firewall.

RESPONDER

Responder.

AGEE

NSXN

HTMLInjection

HTML Injection.

ModelID

Model Number ID.

push

NetScaler Push.

WIonNS

WI on NS.

AppFlow

AppFlow.

CloudBridge

CloudBridge.

CloudBridgeAppliance

CloudExtenderAppliance

ISIS

ISIS Routing.

Cluster

Clustering

CH

Call Home.

AppQoE

AppQoS

APPFLOWICA

Appflow for ICA

isStandardLic

Standard License

isEnterpriseLic

Enterprise License

isPlatinumLic

Platinum License

vPath

Vpath

HWSSL

HW SSL

ns limitIdentifier

Sep 22, 2015

The following operations can be performed on "ns limitIdentifier":

add | rm | set | unset | show | stat

add ns limitIdentifier

Adds a limit identifier to check if the amount of traffic exceeds a specified value, within a particular time interval.

Synopsys

add ns limitIdentifier <limitIdentifier> [-threshold <positive_integer>] [-timeSlice <positive_integer>] [-mode <mode> [-limitType (BURSTY | SMOOTH)]] [-selectorName <string>] [-maxBandwidth <positive_integer>] [-trapsInTimeSlice <positive_integer>]

Arguments

limitIdentifier

Name for a rate limit identifier. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Reserved words must not be used.

threshold

Maximum number of requests that are allowed in the given timeslice when requests (mode is set as REQUEST_RATE) are tracked per timeslice.

When connections (mode is set as CONNECTION) are tracked, it is the total number of connections that would be let through.

Default value: 1

Minimum value: 1

timeSlice

Time interval, in milliseconds, specified in multiples of 10, during which requests are tracked to check if they cross the threshold. This argument is needed only when the mode is set to REQUEST_RATE.

Default value: 1000

Minimum value: 10

mode

Defines the type of traffic to be tracked.

* REQUEST_RATE - Tracks requests/timeslice.

* CONNECTION - Tracks active transactions.

Examples

1. To permit 20 requests in 10 ms and 2 traps in 10 ms:

add limitidentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200

2. To permit 50 requests in 10 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType smooth

3. To permit 1 request in 40 ms:

set limitidentifier limit_req -mode request_rate -timeslice 2000 -Threshold 50 -limitType smooth

4. To permit 1 request in 200 ms and 1 trap in 130 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5 -limitType smooth -trapsInTimeSlice 8

5. To permit 5000 requests in 1000 ms and 200 traps in 1000 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType BURSTY

Possible values: CONNECTION, REQUEST_RATE, NONE

Default value: PEMGMT_RLT_MODE_REQ_RATE

limitType

Smooth or bursty request type.

* SMOOTH - When you want the permitted number of requests in a given interval of time to be spread evenly across the timeslice

* BURSTY - When you want the permitted number of requests to exhaust the quota anytime within the timeslice.

This argument is needed only when the mode is set to REQUEST_RATE.

Possible values: BURSTY, SMOOTH

Default value: PEMGMT_RLT_REQ_RATE_TYPE_BURSTY

selectorName

Name of the rate limit selector. If this argument is NULL, rate limiting will be applied on all traffic received by the virtual server or the NetScaler (depending on whether the limit identifier is bound to a virtual server or globally) without any filtering.

maxBandwidth

Maximum bandwidth permitted, in kbps.

Maximum value: 4294967287

trapsInTimeSlice

Number of traps to be sent in the timeslice configured. A value of 0 indicates that traps are disabled.

Maximum value: 65535

Example

```
add ns limitIdentifier limit_id -threshold 2 -timeSlice 5000 -mode CONNECTION  -selectorName sel_1 -maxBandwidth 24 -trapsInTimeSlice 8
```

---

rm ns limitIdentifier

Removes a rate limit identifier from the appliance.

Synopsys

```
rm ns limitIdentifier <limitIdentifier>
```

Arguments

limitIdentifier

Name of the rate limit identifier to be removed.

Example

```
rm ns limitIdentifier limit_id
```

---

set ns limitIdentifier

Modifies the attributes of a rate limit identifier.

Synopsys

```
set ns limitIdentifier <limitIdentifier> [-threshold <positive_integer>] [-timeSlice <positive_integer>] [-mode <mode> [-limitType ( BURSTY | SMOOTH )]] [-selectorName <string>] [-maxBandwidth <positive_integer>] [-trapsInTimeSlice <positive_integer>]
```

Arguments

limitIdentifier

Name of the rate limit identifier to be modified.

threshold

Maximum number of requests that are allowed in the given timeslice when requests (mode is set as REQUEST_RATE) are tracked per timeslice.

When connections (mode is set as CONNECTION) are tracked, it is the total number of connections that would be let through.

Default value: 1

Minimum value: 1

timeSlice

Time interval, in milliseconds, specified in multiples of 10, during which requests are tracked to check if they cross the threshold. This argument is needed only when the mode is set to REQUEST_RATE.

Default value: 1000

Minimum value: 10

mode

Defines the type of traffic to be tracked.

* REQUEST_RATE - Tracks requests/timeslice.

* CONNECTION - Tracks active transactions.

Examples

1. To permit 20 requests in 10 ms and 2 traps in 10 ms:

add limitidentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200

2. To permit 50 requests in 10 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType smooth

3. To permit 1 request in 40 ms:

set limitidentifier limit_req -mode request_rate -timeslice 2000 -Threshold 50 -limitType smooth

4. To permit 1 request in 200 ms and 1 trap in 130 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5 -limitType smooth -trapsInTimeSlice 8

5. To permit 5000 requests in 1000 ms and 200 traps in 1000 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType BURSTY

Possible values: CONNECTION, REQUEST_RATE, NONE

Default value: PEMGMT_RLT_MODE_REQ_RATE

selectorName

Name of the rate limit selector. If this argument is NULL, rate limiting will be applied on all traffic received by the virtual server or the NetScaler (depending on whether the limit identifier is bound to a virtual server or globally) without any filtering.

maxBandwidth

Maximum bandwidth permitted, in kbps.

Maximum value: 4294967287

trapsInTimeSlice

Number of traps to be sent in the timeslice configured. A value of 0 indicates that traps are disabled.

Maximum value: 65535

Example

```
set ns limitIdentifier limit_id -threshold 2 -timeSlice 5000 -mode CONNECTION  -selectorName sel_1 -maxBandwidth 24 -trapsInTimeSlice 8
```

unset ns limitIdentifier

Use this command to remove ns limitIdentifier settings.Refer to the set ns limitIdentifier command for meanings of the arguments.

Synopsys

unset ns limitIdentifier <limitIdentifier> [-selectorName] [-threshold] [-timeSlice] [-mode] [-limitType] [-maxBandwidth] [-trapsInTimeSlice]

---

show ns limitIdentifier

Displays information about a rate limit identifier.

Synopsys

show ns limitIdentifier [<limitIdentifier>]

Arguments

limitIdentifier

Name of the rate limit identifier about which to display information. If a name is not provided, information about all rate limit identifiers is shown.

summary

fullValues

format

level

Outputs

ngname

Nodegroup name to which this identifier belongs to.

threshold

Maximum number of requests that are allowed in the given timeslice when requests (mode is set as REQUEST_RATE) are tracked per timeslice.

When connections (mode is set as CONNECTION) are tracked, it is the total number of connections that would be let through.

timeSlice

Defines the time interval in msecs specified in multiples of 10 msec during which the requests are tracked to see if they cross the threshold. It is used and displayed only when the mode is REQUEST_RATE while tracking request rate and for defining the trap timeslice.

mode

Defines the type of traffic to be tracked.

* REQUEST_RATE - Tracks requests/timeslice.

* CONNECTION - Tracks active transactions.

Examples

1. To permit 20 requests in 10 ms and 2 traps in 10 ms:

add limitidentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200

2. To permit 50 requests in 10 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType smooth

3. To permit 1 request in 40 ms:

set limitidentifier limit_req -mode request_rate -timeslice 2000 -Threshold 50 -limitType smooth

4. To permit 1 request in 200 ms and 1 trap in 130 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5 -limitType smooth -trapsInTimeSlice 8

5. To permit 5000 requests in 1000 ms and 200 traps in 1000 ms:

set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType BURSTY

limitType

Smooth or bursty request type.

* SMOOTH - When you want the permitted number of requests in a given interval of time to be spread evenly across the timeslice

* BURSTY - When you want the permitted number of requests to exhaust the quota anytime within the timeslice.

This argument is needed only when the mode is set to REQUEST_RATE.

selectorName

Name of the rate limit selector. If this argument is NULL, rate limiting will be applied on all traffic received by the virtual server or the NetScaler (depending on whether the limit identifier is bound to a virtual server or globally) without any filtering.

stateflag

This is used internally to identify ip addresses returned.

hits

The number of times this identifier was evaluated.

drop

The number of times action was taken.

rule

Rule.

time

Time interval considered for rate limiting

total

Maximum number of requests permitted in the computed timeslice

maxBandwidth

The maximum bandwidth in kbps permitted

trapsInTimeSlice

The maximum bandwidth permitted in kbps

trapsComputedInTimeSlice

The number of traps that would be sent in the timeslice configured.

computedTrapTimeSlice

The time interval computed for sending traps.

referenceCount

Total number of transactions pointing to this entry.

devno

count

Example

```
show ns limitIdentifier limit_id
```

stat ns limitIdentifier

Display statistics of a identifier.

Synopsys

stat ns limitIdentifier [<name> [<pattern> ...]] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)] [-sortBy Hits [<sortOrder>]]

Arguments

name

The name of the identifier.

pattern

Pattern for the selector field, ? means field is required, * means field value does not matter, anything else is a regular pattern

clearstats

Clear the statsistics / counters

Possible values: basic, full

sortBy

use this argument to sort by specific key

Possible values: Hits

Outputs

count

devno

stateflag

Outputs

Rate Limit Identifier Hits (Hits)

Total hits.

Rate Limit Identifier Drops (Drops)

Total drops

Rate Limit Session Hits (Hits)

Total hits.

ns limitSelector

Sep 22, 2015

The following operations can be performed on "ns limitSelector":

add | rm | set | unset | show

add ns limitSelector

NOTE: This command is deprecated.Replaced by "stream selector"

Synopsys

Arguments

selectorName

rule

---

rm ns limitSelector

NOTE: This command is deprecated.Replaced by "stream selector"

Synopsys

Arguments

selectorName

---

set ns limitSelector

NOTE: This command is deprecated.

Synopsys

Arguments

selectorName

rule

---

unset ns limitSelector

Use this command to remove ns limitSelector settings.Refer to the set ns limitSelector command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

show ns limitSelector

NOTE: This command is deprecated.Replaced by "stream selector"

Synopsys

Arguments

selectorName

summary

fullValues

format

level

Outputs

rule

stateflag

devno

count

ns limitSessions

Sep 22, 2015

The following operations can be performed on "ns limitSessions":

show | clear

show ns limitSessions

Displays the rate limit sessions available on the appliance.

Synopsys

show ns limitSessions <limitIdentifier> [-detail]

Arguments

limitIdentifier

Name of the rate limit identifier for which to display the sessions.

detail

Show the individual hash values.

Outputs

timeout

The time remaining on the session before a flush can be attempted. If active transactions are present the session will not be flushed

hits

The number of times this entry was hit.

drop

The number of times action was taken.

number

The hash of the matched selectlets.

name

The string formed by gathering selectlet values.

unit

Total computed hash of the matched selectlets.

flags

Used internally to identify ip addresses.

referenceCount

Total number of transactions pointing to this entry. Its the sum total of the connection and bandwidth references

maxBandwidth

The current bandwidth

SelectorIPV61

First IPV6 address gathered.

SelectorIPV62

Second IPV6 address gathered.

flag

Used internally to identify ipv6 addresses.

devno

count

stateflag

---

clear ns limitSessions

Clears the rate limit sessions available on the appliance.

Synopsys

clear ns limitSessions <limitIdentifier>

Arguments

limitIdentifier

Name of the rate limit identifier for which the sessions must be cleared.

---

ns memory

Sep 22, 2015

The following operations can be performed on "ns memory":

stat ns memory

Displays memory statistics of NetScaler features.

Synopsys

stat ns memory [<pool>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

pool

Feature name for which to display memory statistics.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

Allocation failure (AllocF)

Memory allocation failure for particular feature.

Percentage of memory allocated (Alloc(%))

Percentage of NetScaler memory used by the feature.

Total memory Allocated (KB) (CurAlloc(KB))

Total current NetScaler memory available for use by the feature, in kilobytes.

ns mode

Sep 22, 2015

The following operations can be performed on "ns mode":

enable | disable | show

enable ns mode

Enables NetScaler mode(s).

Synopsys

enable ns mode <Mode> ...

Arguments

Mode

Mode to be enabled. Multiple modes can be specified by providing a blank space between each mode.

Example

This CLI command enables the system's client keep-alive feature: enable ns mode CKA

---

disable ns mode

Disables NetScaler mode(s).

Synopsys

disable ns mode <Mode> ...

Arguments

Mode

Mode to be disabled. Multiple modes can be specified by providing a blank space between each mode.

Example

This example shows the command to disable the system's client keep-alive feature: disable ns mode CKA

---

show ns mode

Displays the current state of NetScaler modes.

Synopsys

show ns mode

Outputs

Mode

Mode to be enabled. Multiple modes can be specified by providing a blank space between each mode.

FR

Fast Ramp.

L2

Layer 2 mode.

USIP

Use Source IP.

CKA

Client Keep-alive.

TCPB

TCP Buffering.

MBF

MAC-based forwarding.

Edge

Edge configuration.

USNIP

Use Subnet IP.

L3

Layer 3 mode (ip forwarding).

PMTUD

Path MTU Discovery.

SRADV

Static Route Advertisement.

DRADV

Direct Route Advertisement.

IRADV

Intranet Route Advertisement.

SRADV6

Ipv6 Static Route Advertisement.

DRADV6

Ipv6 Direct Route Advertisement.

BridgeBPDUs

BPDU Bridging Mode.

ns ns.conf

Sep 22, 2015

The following operations can be performed on "ns ns.conf":

show ns ns.conf

Displays the saved configurations.

Synopsys

show ns ns.conf

Outputs

textBlob

Text of the last saved configuration.

ns param

Sep 22, 2015

The following operations can be performed on "ns param":

set | unset | show

set ns param

Sets the parameters of the NetScaler appliance.

Synopsys

set ns param [-httpPort <port> ...] [-maxConn <positive_integer>] [-maxReq <positive_integer>] [-cip (ENABLED | DISABLED) <cipHeader>] [-cookieversion (0 | 1)] [-secureCookie (ENABLED | DISABLED)] [-pmtuMin <positive_integer>] [-pmtuTimeout <mins>] [-ftpPortRange <int[-int]>] [-crPortRange <int[-int]>] [-timezone <timezone>] [-grantQuotaMaxClient <positive_integer>] [-exclusiveQuotaMaxClient <positive_integer>] [-grantQuotaSpillOver <positive_integer>] [-exclusiveQuotaSpillOver <positive_integer>] [-useproxyport (ENABLED | DISABLED)] [-internaluserlogin (ENABLED | DISABLED)] [-aftpAllowRandomSourcePort (ENABLED | DISABLED)] [-icaPorts <port> ...]

Arguments

httpPort

HTTP ports on the web server. This allows the system to perform connection off-load for any client request that has a destination port matching one of these configured ports.

Minimum value: 1

maxConn

Maximum number of connections that will be made from the appliance to the web server(s) attached to it. The value entered here is applied globally to all attached servers.

Maximum value: 4294967294

maxReq

Maximum number of requests that the system can pass on a particular connection between the appliance and a server attached to it. Setting this value to 0 allows an unlimited number of requests to be passed. This value is overridden by the maximum number of requests configured on the individual service.

Maximum value: 65535

cip

Enable or disable the insertion of the actual client IP address into the HTTP header request passed from the client to one, some, or all servers attached to the system. The passed address can then be accessed through a minor modification to the server.

* If the CIP header is specified, it will be used as the client IP header.

* If the CIP header is not specified, the value that has been set will be used as the client IP header.

Possible values: ENABLED, DISABLED

cookieversion

Version of the cookie inserted by the system.

Possible values: 0, 1

secureCookie

Enable or disable secure flag for persistence cookie.

Possible values: ENABLED, DISABLED

Default value: ENABLED

pmtuMin

Minimum path MTU value that NetScaler will process in the ICMP fragmentation needed message. If the ICMP message contains a value less than this value, then this value is used instead.

Default value: 576

Minimum value: 168

Maximum value: 1500

pmtuTimeout

Interval, in minutes, for flushing the PMTU entries.

Default value: 10

Minimum value: 1

Maximum value: 1440

ftpPortRange

Minimum and maximum port (port range) that FTP services are allowed to use.

Minimum value: 1024

Maximum value: 64000

crPortRange

Port range for cache redirection services.

Minimum value: 1

Maximum value: 65535

timezone

Time zone for the NetScaler appliance. Name of the time zone should be specified as argument.

Possible values: CoordinatedUniversalTime, GMT+01:00-CET-Europe/Andorra, GMT+04:00-GST-Asia/Dubai, GMT+04:30-AFT-Asia/Kabul, GMT-04:00-AST-America/Antigua, GMT-04:00-AST-America/Anguilla, GMT+01:00-CET-Europe/Tirane, GMT+04:00-AMT-Asia/Yerevan, GMT+01:00-WAT-Africa/Luanda, GMT+13:00-NZDT-Antarctica/McMurdo, GMT+13:00-NZDT-Antarctica/South_Pole, GMT-03:00-ROTT-Antarctica/Rothera, GMT-04:00-CLT-Antarctica/Palmer, GMT+05:00-MAWT-Antarctica/Mawson, GMT+07:00-DAVT-Antarctica/Davis, GMT+08:00-WST-Antarctica/Casey, GMT+06:00-VOST-Antarctica/Vostok, GMT+10:00-DDUT-Antarctica/DumontDUrville, GMT+03:00-SYOT-Antarctica/Syowa, GMT+11:00-MIST-Antarctica/Macquarie, GMT-03:00-ART-America/Argentina/Buenos_Aires, GMT-03:00-ART-America/Argentina/Cordoba, GMT-03:00-ART-America/Argentina/Salta, GMT-03:00-ART-America/Argentina/Jujuy, GMT-03:00-ART-America/Argentina/Tucuman, GMT-03:00-ART-America/Argentina/Catamarca, GMT-03:00-ART-America/Argentina/La_Rioja, GMT-03:00-ART-America/Argentina/San_Juan, GMT-03:00-ART-America/Argentina/Mendoza, GMT-03:00-WARST-America/Argentina/San_Luis, GMT-03:00-ART-America/Argentina/Rio_Gallegos, GMT-03:00-ART-America/Argentina/Ushuaia, GMT-11:00-SST-Pacific/Pago_Pago, GMT+01:00-CET-Europe/Vienna, GMT+11:00-LHST-Australia/Lord_Howe, GMT+11:00-EST-Australia/Hobart, GMT+11:00-EST-Australia/Currie, GMT+11:00-EST-Australia/Melbourne, GMT+11:00-EST-Australia/Sydney, GMT+10:30-CST-Australia/Broken_Hill, GMT+10:00-EST-Australia/Brisbane, GMT+10:00-EST-Australia/Lindeman, GMT+10:30-CST-Australia/Adelaide, GMT+09:30-CST-Australia/Darwin, GMT+08:00-WST-Australia/Perth, GMT+08:45-CWST-Australia/Eucla, GMT-04:00-AST-America/Aruba, GMT+02:00-EET-Europe/Mariehamn, GMT+04:00-AZT-Asia/Baku, GMT+01:00-CET-Europe/Sarajevo, GMT-04:00-AST-America/Barbados, GMT+06:00-BDT-Asia/Dhaka, GMT+01:00-CET-Europe/Brussels, GMT+00:00-GMT-Africa/Ouagadougou, GMT+02:00-EET-Europe/Sofia, GMT+03:00-AST-Asia/Bahrain, GMT+02:00-CAT-Africa/Bujumbura, GMT+01:00-WAT-Africa/Porto-Novo, GMT-04:00-AST-America/St_Barthelemy, GMT-03:00-ADT-Atlantic/Bermuda, GMT+08:00-BNT-Asia/Brunei, GMT-04:00-BOT-America/La_Paz, GMT-02:00-FNT-America/Noronha, GMT-03:00-BRT-America/Belem, GMT-03:00-BRT-America/Fortaleza, GMT-03:00-BRT-America/Recife, GMT-03:00-BRT-America/Araguaina, GMT-03:00-BRT-America/Maceio, GMT-03:00-BRT-America/Bahia, GMT-03:00-BRT-America/Sao_Paulo, GMT-04:00-AMT-America/Campo_Grande, GMT-04:00-AMT-America/Cuiaba, GMT-03:00-BRT-America/Santarem, GMT-04:00-AMT-America/Porto_Velho, GMT-04:00-AMT-America/Boa_Vista, GMT-04:00-AMT-America/Manaus, GMT-04:00-AMT-America/Eirunepe, GMT-04:00-AMT-America/Rio_Branco, GMT-04:00-EDT-America/Nassau, GMT+06:00-BTT-Asia/Thimphu, GMT+02:00-CAT-Africa/Gaborone, GMT+03:00-FET-Europe/Minsk, GMT-06:00-CST-America/Belize, GMT-02:30-NDT-America/St_Johns, GMT-03:00-ADT-America/Halifax, GMT-03:00-ADT-America/Glace_Bay, GMT-03:00-ADT-America/Moncton, GMT-03:00-ADT-America/Goose_Bay, GMT-04:00-AST-America/Blanc-Sablon, GMT-04:00-EDT-America/Montreal, GMT-04:00-EDT-America/Toronto, GMT-04:00-EDT-America/Nipigon, GMT-04:00-EDT-America/Thunder_Bay, GMT-04:00-EDT-America/Iqaluit, GMT-04:00-EDT-America/Pangnirtung, GMT-05:00-CDT-America/Resolute, GMT-05:00-EST-America/Atikokan, GMT-05:00-CDT-America/Rankin_Inlet, GMT-05:00-CDT-America/Winnipeg, GMT-05:00-CDT-America/Rainy_River, GMT-06:00-CST-America/Regina, GMT-06:00-CST-America/Swift_Current, GMT-06:00-MDT-America/Edmonton, GMT-06:00-MDT-America/Cambridge_Bay, GMT-06:00-MDT-America/Yellowknife, GMT-06:00-MDT-America/Inuvik, GMT-07:00-MST-America/Dawson_Creek, GMT-07:00-PDT-America/Vancouver, GMT-07:00-PDT-America/Whitehorse, GMT-07:00-PDT-America/Dawson, GMT+06:30-CCT-Indian/Cocos, GMT+01:00-WAT-Africa/Kinshasa, GMT+02:00-CAT-Africa/Lubumbashi, GMT+01:00-WAT-Africa/Bangui, GMT+01:00-WAT-Africa/Brazzaville, GMT+01:00-CET-Europe/Zurich, GMT+00:00-GMT-Africa/Abidjan, GMT-10:00-CKT-Pacific/Rarotonga, GMT-04:00-CLT-America/Santiago, GMT-06:00-EAST-Pacific/Easter, GMT+01:00-WAT-Africa/Douala, GMT+08:00-CST-Asia/Shanghai, GMT+08:00-CST-Asia/Harbin, GMT+08:00-CST-Asia/Chongqing, GMT+08:00-CST-Asia/Urumqi, GMT+08:00-CST-Asia/Kashgar, GMT-05:00-COT-America/Bogota, GMT-06:00-CST-America/Costa_Rica, GMT-04:00-CDT-America/Havana, GMT-01:00-CVT-Atlantic/Cape_Verde, GMT+07:00-CXT-Indian/Christmas, GMT+02:00-EET-Asia/Nicosia, GMT+01:00-CET-

Europe/Prague, GMT+01:00-CET-Europe/Berlin, GMT+03:00-EAT-Africa/Djibouti, GMT+01:00-CET-Europe/Copenhagen, GMT-04:00-AST-America/Dominica, GMT-04:00-AST-America/Santo_Domingo, GMT+01:00-CET-Africa/Algiers, GMT-05:00-ECT-America/Guayaquil, GMT-06:00-GALT-Pacific/Galapagos, GMT+02:00-EET-Europe/Tallinn, GMT+02:00-EET-Africa/Cairo, GMT+00:00-WET-Africa/El_Aaiun, GMT+03:00-EAT-Africa/Asmara, GMT+01:00-CET-Europe/Madrid, GMT+01:00-CET-Africa/Ceuta, GMT+00:00-WET-Atlantic/Canary, GMT+03:00-EAT-Africa/Addis_Ababa, GMT+02:00-EET-Europe/Helsinki, GMT+12:00-FJT-Pacific/Fiji, GMT-03:00-FKST-Atlantic/Stanley, GMT+10:00-CHUT-Pacific/Chuuk, GMT+11:00-PONT-Pacific/Pohnpei, GMT+11:00-KOST-Pacific/Kosrae, GMT+00:00-WET-Atlantic/Faroe, GMT+01:00-CET-Europe/Paris, GMT+01:00-WAT-Africa/Libreville, GMT+00:00-GMT-Europe/London, GMT-04:00-AST-America/Grenada, GMT+04:00-GET-Asia/Tbilisi, GMT-03:00-GFT-America/Cayenne, GMT+00:00-GMT-Europe/Guernsey, GMT+00:00-GMT-Africa/Accra, GMT+01:00-CET-Europe/Gibraltar, GMT-03:00-WGT-America/Godthab, GMT+00:00-GMT-America/Danmarkshavn, GMT-01:00-EGT-America/Scoresbysund, GMT-03:00-ADT-America/Thule, GMT+00:00-GMT-Africa/Banjul, GMT+00:00-GMT-Africa/Conakry, GMT-04:00-AST-America/Guadeloupe, GMT+01:00-WAT-Africa/Malabo, GMT+02:00-EET-Europe/Athens, GMT-02:00-GST-Atlantic/South_Georgia, GMT-06:00-CST-America/Guatemala, GMT+10:00-ChST-Pacific/Guam, GMT+00:00-GMT-Africa/Bissau, GMT-04:00-GYT-America/Guyana, GMT+08:00-HKT-Asia/Hong_Kong, GMT-06:00-CST-America/Tegucigalpa, GMT+01:00-CET-Europe/Zagreb, GMT-05:00-EST-America/Port-au-Prince, GMT+01:00-CET-Europe/Budapest, GMT+07:00-WIT-Asia/Jakarta, GMT+07:00-WIT-Asia/Pontianak, GMT+08:00-CIT-Asia/Makassar, GMT+09:00-EIT-Asia/Jayapura, GMT+00:00-GMT-Europe/Dublin, GMT+02:00-IST-Asia/Jerusalem, GMT+00:00-GMT-Europe/Isle_of_Man, GMT+05:30-IST-Asia/Kolkata, GMT+06:00-IOT-Indian/Chagos, GMT+03:00-AST-Asia/Baghdad, GMT+03:30-IRST-Asia/Tehran, GMT+00:00-GMT-Atlantic/Reykjavik, GMT+01:00-CET-Europe/Rome, GMT+00:00-GMT-Europe/Jersey, GMT-05:00-EST-America/Jamaica, GMT+02:00-EET-Asia/Amman, GMT+09:00-JST-Asia/Tokyo, GMT+03:00-EAT-Africa/Nairobi, GMT+06:00-KGT-Asia/Bishkek, GMT+07:00-ICT-Asia/Phnom_Penh, GMT+12:00-GILT-Pacific/Tarawa, GMT+13:00-PHOT-Pacific/Enderbury, GMT+14:00-LINT-Pacific/Kiritimati, GMT+03:00-EAT-Indian/Comoro, GMT-04:00-AST-America/St_Kitts, GMT+09:00-KST-Asia/Pyongyang, GMT+09:00-KST-Asia/Seoul, GMT+03:00-AST-Asia/Kuwait, GMT-05:00-EST-America/Cayman, GMT+06:00-ALMT-Asia/Almaty, GMT+06:00-QYZT-Asia/Qyzylorda, GMT+05:00-AQTT-Asia/Aqtobe, GMT+05:00-AQTT-Asia/Aqtau, GMT+05:00-ORAT-Asia/Oral, GMT+07:00-ICT-Asia/Vientiane, GMT+02:00-EET-Asia/Beirut, GMT-04:00-AST-America/St_Lucia, GMT+01:00-CET-Europe/Vaduz, GMT+05:30-IST-Asia/Colombo, GMT+00:00-GMT-Africa/Monrovia, GMT+02:00-SAST-Africa/Maseru, GMT+02:00-EET-Europe/Vilnius, GMT+01:00-CET-Europe/Luxembourg, GMT+02:00-EET-Europe/Riga, GMT+02:00-EET-Africa/Tripoli, GMT+00:00-WET-Africa/Casablanca, GMT+01:00-CET-Europe/Monaco, GMT+02:00-EET-Europe/Chisinau, GMT+01:00-CET-Europe/Podgorica, GMT-04:00-AST-America/Marigot, GMT+03:00-EAT-Indian/Antananarivo, GMT+12:00-MHT-Pacific/Majuro, GMT+12:00-MHT-Pacific/Kwajalein, GMT+01:00-CET-Europe/Skopje, GMT+00:00-GMT-Africa/Bamako, GMT+06:30-MMT-Asia/Rangoon, GMT+08:00-ULAT-Asia/Ulaanbaatar, GMT+07:00-HOVT-Asia/Hovd, GMT+08:00-CHOT-Asia/Choibalsan, GMT+08:00-CST-Asia/Macau, GMT+10:00-ChST-Pacific/Saipan, GMT-04:00-AST-America/Martinique, GMT+00:00-GMT-Africa/Nouakchott, GMT-04:00-AST-America/Montserrat, GMT+01:00-CET-Europe/Malta, GMT+04:00-MUT-Indian/Mauritius, GMT+05:00-MVT-Indian/Maldives, GMT+02:00-CAT-Africa/Blantyre, GMT-06:00-CST-America/Mexico_City, GMT-06:00-CST-America/Cancun, GMT-06:00-CST-America/Merida, GMT-06:00-CST-America/Monterrey, GMT-05:00-CDT-America/Matamoros, GMT-07:00-MST-America/Mazatlan, GMT-07:00-MST-America/Chihuahua, GMT-06:00-MDT-America/Ojinaga, GMT-07:00-MST-America/Hermosillo, GMT-07:00-PDT-America/Tijuana, GMT-08:00-PST-America/Santa_Isabel, GMT-06:00-CST-America/Bahia_Banderas, GMT+08:00-MYT-Asia/Kuala_Lumpur, GMT+08:00-MYT-Asia/Kuching, GMT+02:00-CAT-Africa/Maputo, GMT+02:00-WAST-Africa/Windhoek, GMT+11:00-NCT-Pacific/Noumea, GMT+01:00-WAT-Africa/Niamey, GMT+11:30-NFT-Pacific/Norfolk, GMT+01:00-WAT-Africa/Lagos, GMT-06:00-CST-America/Managua, GMT+01:00-CET-Europe/Amsterdam, GMT+01:00-CET-Europe/Oslo, GMT+05:45-NPT-Asia/Kathmandu, GMT+12:00-NRT-

Pacific/Nauru, GMT-11:00-NUT-Pacific/Niue, GMT+13:00-NZDT-Pacific/Auckland, GMT+13:45-CHADT-Pacific/Chatham, GMT+04:00-GST-Asia/Muscat, GMT-05:00-EST-America/Panama, GMT-05:00-PET-America/Lima, GMT-10:00-TAHT-Pacific/Tahiti, GMT-09:30-MART-Pacific/Marquesas, GMT-09:00-GAMT-Pacific/Gambier, GMT+10:00-PGT-Pacific/Port_Moresby, GMT+08:00-PHT-Asia/Manila, GMT+05:00-PKT-Asia/Karachi, GMT+01:00-CET-Europe/Warsaw, GMT-02:00-PMDT-America/Miquelon, GMT-08:00-PST-Pacific/Pitcairn, GMT-04:00-AST-America/Puerto_Rico, GMT+02:00-EET-Asia/Gaza, GMT+02:00-EET-Asia/Hebron, GMT+00:00-WET-Europe/Lisbon, GMT+00:00-WET-Atlantic/Madeira, GMT-01:00-AZOT-Atlantic/Azores, GMT+09:00-PWT-Pacific/Palau, GMT-03:00-PYST-America/Asuncion, GMT+03:00-AST-Asia/Qatar, GMT+04:00-RET-Indian/Reunion, GMT+02:00-EET-Europe/Bucharest, GMT+01:00-CET-Europe/Belgrade, GMT+03:00-FET-Europe/Kaliningrad, GMT+04:00-MSK-Europe/Moscow, GMT+04:00-VOLT-Europe/Volgograd, GMT+04:00-SAMT-Europe/Samara, GMT+06:00-YEKT-Asia/Yekaterinburg, GMT+07:00-OMST-Asia/Omsk, GMT+07:00-NOVT-Asia/Novosibirsk, GMT+07:00-NOVT-Asia/Novokuznetsk, GMT+08:00-KRAT-Asia/Krasnoyarsk, GMT+09:00-IRKT-Asia/Irkutsk, GMT+10:00-YAKT-Asia/Yakutsk, GMT+11:00-VLAT-Asia/Vladivostok, GMT+11:00-SAKT-Asia/Sakhalin, GMT+12:00-MAGT-Asia/Magadan, GMT+12:00-PETT-Asia/Kamchatka, GMT+12:00-ANAT-Asia/Anadyr, GMT+02:00-CAT-Africa/Kigali, GMT+03:00-AST-Asia/Riyadh, GMT+11:00-SBT-Pacific/Guadalcanal, GMT+04:00-SCT-Indian/Mahe, GMT+03:00-EAT-Africa/Khartoum, GMT+01:00-CET-Europe/Stockholm, GMT+08:00-SGT-Asia/Singapore, GMT+00:00-GMT-Atlantic/St_Helena, GMT+01:00-CET-Europe/Ljubljana, GMT+01:00-CET-Arctic/Longyearbyen, GMT+01:00-CET-Europe/Bratislava, GMT+00:00-GMT-Africa/Freetown, GMT+01:00-CET-Europe/San_Marino, GMT+00:00-GMT-Africa/Dakar, GMT+03:00-EAT-Africa/Mogadishu, GMT-03:00-SRT-America/Paramaribo, GMT+00:00-GMT-Africa/Sao_Tome, GMT-06:00-CST-America/El_Salvador, GMT+02:00-EET-Asia/Damascus, GMT+02:00-SAST-Africa/Mbabane, GMT-04:00-EDT-America/Grand_Turk, GMT+01:00-WAT-Africa/Ndjamena, GMT+05:00-TFT-Indian/Kerguelen, GMT+00:00-GMT-Africa/Lome, GMT+07:00-ICT-Asia/Bangkok, GMT+05:00-TJT-Asia/Dushanbe, GMT-10:00-TKT-Pacific/Fakaofo, GMT+09:00-TLT-Asia/Dili, GMT+05:00-TMT-Asia/Ashgabat, GMT+01:00-CET-Africa/Tunis, GMT+13:00-TOT-Pacific/Tongatapu, GMT+02:00-EET-Europe/Istanbul, GMT-04:00-AST-America/Port_of_Spain, GMT+12:00-TVT-Pacific/Funafuti, GMT+08:00-CST-Asia/Taipei, GMT+03:00-EAT-Africa/Dar_es_Salaam, GMT+02:00-EET-Europe/Kiev, GMT+02:00-EET-Europe/Uzhgorod, GMT+02:00-EET-Europe/Zaporozhye, GMT+02:00-EET-Europe/Simferopol, GMT+03:00-EAT-Africa/Kampala, GMT-10:00-HST-Pacific/Johnston, GMT-11:00-SST-Pacific/Midway, GMT+12:00-WAKT-Pacific/Wake, GMT-04:00-EDT-America/New_York, GMT-04:00-EDT-America/Detroit, GMT-04:00-EDT-America/Kentucky/Louisville, GMT-04:00-EDT-America/Kentucky/Monticello, GMT-04:00-EDT-America/Indiana/Indianapolis, GMT-04:00-EDT-America/Indiana/Vincennes, GMT-04:00-EDT-America/Indiana/Winamac, GMT-04:00-EDT-America/Indiana/Marengo, GMT-04:00-EDT-America/Indiana/Petersburg, GMT-04:00-EDT-America/Indiana/Vevay, GMT-05:00-CDT-America/Chicago, GMT-05:00-CDT-America/Indiana/Tell_City, GMT-05:00-CDT-America/Indiana/Knox, GMT-05:00-CDT-America/Menominee, GMT-05:00-CDT-America/North_Dakota/Center, GMT-05:00-CDT-America/North_Dakota/New_Salem, GMT-05:00-CDT-America/North_Dakota/Beulah, GMT-06:00-MDT-America/Denver, GMT-06:00-MDT-America/Boise, GMT-06:00-MDT-America/Shiprock, GMT-07:00-MST-America/Phoenix, GMT-07:00-PDT-America/Los_Angeles, GMT-08:00-AKDT-America/Anchorage, GMT-08:00-AKDT-America/Juneau, GMT-08:00-AKDT-America/Sitka, GMT-08:00-AKDT-America/Yakutat, GMT-08:00-AKDT-America/Nome, GMT-09:00-HADT-America/Adak, GMT-08:00-MeST-America/Metlakatla, GMT-10:00-HST-Pacific/Honolulu, GMT-03:00-UYT-America/Montevideo, GMT+05:00-UZT-Asia/Samarkand, GMT+05:00-UZT-Asia/Tashkent, GMT+01:00-CET-Europe/Vatican, GMT-04:00-AST-America/St_Vincent, GMT-04:30-VET-America/Caracas, GMT-04:00-AST-America/Tortola, GMT-04:00-AST-America/St_Thomas, GMT+07:00-ICT-Asia/Ho_Chi_Minh, GMT+11:00-VUT-Pacific/Efate, GMT+12:00-WFT-Pacific/Wallis, GMT+14:00-WSDT-Pacific/Apia, GMT+03:00-AST-Asia/Aden, GMT+03:00-EAT-Indian/Mayotte, GMT+02:00-SAST-Africa/Johannesburg, GMT+02:00-CAT-Africa/Lusaka, GMT+02:00-CAT-Africa/Harare

grantQuotaMaxClient

Percentage of shared quota to be granted at a time for maxClient.

Default value: 10

Maximum value: 100

exclusiveQuotaMaxClient

Percentage of maxClient to be given to PEs.

Default value: 80

Maximum value: 100

grantQuotaSpillOver

Percentage of shared quota to be granted at a time for spillover.

Default value: 10

Maximum value: 100

exclusiveQuotaSpillOver

Percentage of maximum limit to be given to PEs.

Default value: 80

Maximum value: 100

useproxyport

Enable/Disable use_proxy_port setting

Possible values: ENABLED, DISABLED

Default value: ENABLED

internaluserlogin

Enables/disables the internal user from logging in to the appliance. Before disabling internal user login, you must have key-based authentication set up on the appliance. The file name for the key pair must be "ns_comm_key".

Possible values: ENABLED, DISABLED

Default value: ENABLED

aftpAllowRandomSourcePort

Allow the FTP server to come from a random source port for active FTP data connections

Possible values: ENABLED, DISABLED

Default value: DISABLED

icaPorts

The ICA ports on the Web server. This allows the system to perform connection off-load for any

client request that has a destination port matching one of these configured ports.

Minimum value: 1

---

unset ns param

Removes the attributes of the NetScaler parameters. Attributes for which a default value is available revert to their default values. Refer to the 'set ns param' command for a description of the parameters..Refer to the set ns param command for meanings of the arguments.

Synopsys

unset ns param [-ftpPortRange] [-crPortRange] [-timezone] [-aftpAllowRandomSourcePort] [-httpPort] [-maxConn] [-maxReq] [-cip] [-cipHeader] [-cookieversion] [-secureCookie] [-pmtuMin] [-pmtuTimeout] [-grantQuotaMaxClient] [-exclusiveQuotaMaxClient] [-grantQuotaSpillOver] [-exclusiveQuotaSpillOver] [-useproxyport] [-internaluserlogin] [-icaPorts]

---

show ns param

Displays the information of the parameters of the NetScaler appliance that were set by using the 'set ns param' command.

Synopsys

show ns param

Arguments

format

level

Outputs

httpPort

The HTTP ports on the Web server.

maxConn

Maximum Number of Connections.

maxReq

Maxmimum Number of requests that can be handled.

cip

Insertion of client IP address into the HTTP header.

cipHeader

The text that will be used as the client IP header.

cookieversion

Version of the cookie inserted by the system.

secureCookie

Enable or disable secure flag for persistence cookie.

pmtuMin

Minimum path MTU value that NetScaler will process in the ICMP fragmentation needed message. If the ICMP message contains a value less than this value, then this value is used instead.

pmtuTimeout

Interval, in minutes, for flushing the PMTU entries.

ftpPortRange

Minimum and maximum port (port range) that FTP services are allowed to use.

crPortRange

Port range for cache redirection services.

timezone

Time zone for the NetScaler appliance. Name of the time zone should be specified as argument.

grantQuotaMaxClient

Percentage of shared quota to be granted at a time for maxClient.

exclusiveQuotaMaxClient

Percentage of maxClient to be given to PEs.

grantQuotaSpillOver

Percentage of shared quota to be granted at a time for spillover.

exclusiveQuotaSpillOver

Percentage of maximum limit to be given to PEs.

useproxyport

Enable/Disable use_proxy_port setting

internaluserlogin

Enables/disables the internal user from logging in to the appliance. Before disabling internal user login, you

must have key-based authentication set up on the appliance. The file name for the key pair must be "ns_comm_key".

aftpAllowRandomSourcePort

Allow the FTP server to come from a random source port for active FTP data connections

icaPorts

The ICA ports on the Web server. This allows the system to perform connection off-load for any client request that has a destination port matching one of these configured ports.

ns pbr

Sep 22, 2015

The following operations can be performed on "ns pbr":

add | rm | set | unset | enable | disable | stat | show

add ns pbr

Adds a policy based route (PBR) to the NetScaler appliance. To commit this operation, you must apply the PBRs. A PBR specifies criteria for selecting outgoing IPv4 packets and, typically, a next hop to which to send the selected packets. For example, you can configure the NetScaler appliance to route outgoing packets from a specific IP address or range to a particular next hop router. Note: The NetScaler appliance process PBRs before processing the RNAT rules.

Synopsys

add ns pbr <name> <action> [-td <positive_integer>] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] ((-nextHop <nextHopVal>) | (-ipTunnel <ipTunnelName>)) [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-state (ENABLED | DISABLED)]

Arguments

name

Name for the PBR. Must begin with an ASCII alphabetic or underscore \\(_\\) character, and must contain only ASCII alphanumeric, underscore, hash \\(\\#\\), period \\(.\\), space, colon \\(:\\), at \\(@\\), equals \\(=\\), and hyphen \\(-\\) characters. Can be changed after the PBR is created.

action

Action to perform on the outgoing IPv4 packets that match the PBR.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

td

Traffic Domain Id.

Maximum value: 4094

srcIP

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destIP

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

nextHop

IP address of the next hop router or the name of the link load balancing virtual server to which to send matching packets if action is set to ALLOW.

If you specify a link load balancing (LLB) virtual server, which can provide a backup if a next hop link fails, first make sure that the next hops bound to the LLB virtual server are actually next hops that are directly connected to the NetScaler appliance. Otherwise, the NetScaler throws an error when you attempt to create the PBR.

ipTunnel

The Tunnel name.

srcMac

MAC address to match against the source MAC address of an outgoing IPv4 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv4 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance compares the PBR only to the outgoing packets on the specified interface. If you do not specify any value, the appliance compares the PBR to the outgoing packets on all interfaces.

priority

Priority of the PBR, which determines the order in which it is evaluated relative to the other PBRs. If you do not specify priorities while creating PBRs, the PBRs are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 81920

msr

Monitor the route specified byte Next Hop parameter. This parameter is not applicable if you specify a link load balancing (LLB) virtual server name with the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

state

Enable or disable the PBR. After you apply the PBRs, the NetScaler appliance compares outgoing packets to the enabled PBRs.

Possible values: ENABLED, DISABLED

Default value: XACLENABLED

Example

```
add ns pbr a allow -srcip 10.102.37.252 -destip 10.10.10.2 -nexthop 11.11.11.2
```

---

rm ns pbr

Removes a PBR from the NetScaler appliance. To commit this operation, you must apply the PBRs.

Synopsys

```
rm ns pbr <name> ...
```

Arguments

name

Name of the PBR that you want to remove.

Example

```
rm ns pbr a
```

---

set ns pbr

Modifies the specified parameters of a PBR. To commit this operation, you must apply the PBRs.

Synopsys

```
set ns pbr <name> [-action ( ALLOW | DENY )] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] ((-nextHop <nextHopVal>) | (-ipTunnel <ipTunnelName>)) [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr ( ENABLED | DISABLED ) [-monitor <string>]]
```

Arguments

name

Name of the PBR whose parameters you want to modify.

action

Action to perform on the outgoing IPv4 packets that match the PBR.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

srcIP

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destIP

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

nextHop

IP address of the next hop router or the name of the link load balancing virtual server to which to send matching packets if action is set to ALLOW.

If you specify a link load balancing (LLB) virtual server, which can provide a backup if a next hop link fails, first make sure that the next hops bound to the LLB virtual server are actually next hops that are directly connected to the NetScaler appliance. Otherwise, the NetScaler throws an error when you attempt to create the PBR.

ipTunnel

The Tunnel name.

srcMac

MAC address to match against the source MAC address of an outgoing IPv4 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv4 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance compares the PBR only to the outgoing packets on the specified interface. If you do not specify any value, the appliance compares the PBR to the outgoing packets on all interfaces.

priority

Priority of the PBR, which determines the order in which it is evaluated relative to the other PBRs. If you do not specify priorities while creating PBRs, the PBRs are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 81920

msr

Monitor the route specified byte Next Hop parameter. This parameter is not applicable if you specify a link load balancing (LLB) virtual server name with the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set ns pbr a -srcPort 50
```

---

unset ns pbr

Resets the attributes of the specified PBR. Attributes for which a default value is available revert to their default values. Refer to the set ns pbr command for descriptions of the parameters..Refer to the set ns pbr command for meanings of the arguments.

Synopsys

```
unset ns pbr <name> [-srcIP] [-srcPort] [-destIP] [-destPort] [-nextHop] [-ipTunnel] [-srcMac] [-protocol] [-vlan] [-interface] [-msr] [-monitor]
```

Example

```
unset ns pbr rule1 -srcPort
```

---

enable ns pbr

Enables a PBR. To commit this operation, you must apply the PBRs. After you apply the PBRs, the NetScaler appliance compares outgoing packets to the enabled PBRs.

Synopsys

```
enable ns pbr <name> ...
```

Arguments

name

Name of PBR that you want to enable.

Example

```
enable ns pbr foo
```

---

disable ns pbr

Disables a PBR. To commit this operation, you must apply the PBRs. After you apply the PBRs, the NetScaler appliance does not compare outgoing packets against the disabled PBRs

Synopsys

```
disable ns pbr <name> ...
```

Arguments

name

Name of PBR that you want to disable.

Example

```
disable ns pbr foo
```

---

stat ns pbr

Displays statistics related to the PBRs. To display statistics of all the PBRs, run the command without any parameters. To display statistics of a particular PBR, specify the name of the PBR.

Synopsys

```
stat ns pbr [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the PBR whose statistics you want the NetScaler appliance to display.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Allow PBR hits (PBRAllow)

Total packets that matched the PBR (Policy-Based Routes) with action ALLOW

Deny PBR hits (PBRDeny)

Total packets that matched the PBR with action DENY

PBR hits (PBRTotHits)

Total packets that matched one of the configured PBR

PBR misses (PBRMiss)

Total packets that did not match any PBR

Hits for this PBR (PBRHits)

Number of times the pbr was hit

Example

stat pbr

---

show ns pbr

Displays settings related to the PBRs. To display settings of all the PBRs, run the command without any parameters. To display settings of a particular PBR, specify the name of the PBR.

Synopsys

show ns pbr [<name>] [-detail]

Arguments

name

Name of the PBR whose details you want the NetScaler appliance to display.

detail

To get a detailed view.

summary

fullValues

format

level

Outputs

action

Action to perform on the outgoing IPv4 packets that match the PBR.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

srcMac

MAC address to match against the source MAC address of an outgoing IPv4 packet.

protocol

The protocol number in IP header or name.

protocolNumber

The protocol number in IP header or name.

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

td

Traffic Domain Id.

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcIPVal

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

vlan

ID of the VLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VLANs.

state

If this route is UP/DOWN.

interface

ID of an interface. The NetScaler appliance compares the PBR only to the outgoing packets on the specified interface. If you do not specify any value, the appliance compares the PBR to the outgoing packets on all interfaces.

hits

The hits of this PBR.

priority

Priority of the PBR, which determines the order in which it is evaluated relative to the other PBRs. If you do not specify priorities while creating PBRs, the PBRs are evaluated in the order in which they are created.

operator

Logical operator.

kernelstate

The commit status of the PBR.

nextHopVal

The Next Hop IP address or gateway name.

ipTunnelName

The iptunnel name where packets need to be forwarded upon.

msr

Whether Monitored Static Route(MSR) is enabled or disabled.

monitor

Name of the monitor, of type PING or ARP, configured on the NetScaler appliance to monitor the route specified by the Next Hop parameter. You must enable the MSR parameter before setting this parameter.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

data

Internal data of this route.

devno

count

stateflag

Example

```
show ns pbr a  Name: a                    Action: ALLOW    Hits: 0      srcIP = 10.102.37.252      destIP = 10.10.10.2
```

ns pbr6

Sep 22, 2015

The following operations can be performed on "ns pbr6":

add | renumber | rm | set | unset | enable | disable | stat | show | clear | apply

add ns pbr6

Adds an IPv6 policy based route (PBR6) to the NetScaler appliance. To commit this operation, you must apply the PBR6s. A PBR6 specifies criteria for selecting outgoing IPv6 packets and, typically, a next hop to which to send the selected packets. For example, you can configure the NetScaler appliance to route outgoing packets from a specific IP address or range to a particular next hop router. Note: The NetScaler appliance process PBR6s before processing the RNAT rules.

Synopsys

add ns pbr6 <name> [-td <positive_integer>] <action> [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-state (ENABLED | DISABLED)] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-nextHop <nextHopVal>] [-nextHopVlan <positive_integer>]

Arguments

name

Name for the PBR6. Must begin with an ASCII alphabetic or underscore \\(_\\) character, and must contain only ASCII alphanumeric, underscore, hash \\(\\#\\), period \\(.\\), space, colon \\(:\\), at \\(@\\), equals \\(=\\), and hyphen \\(-\\) characters. Can be changed after the PBR6 is created.

td

Traffic Domain Id.

Maximum value: 4094

action

Action to perform on the outgoing IPv6 packets that match the PBR6.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

srcIPv6

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

srcPort

Port number or range of port numbers to match against the source port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destIPv6

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destPort

Port number or range of port numbers to match against the destination port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcMac

MAC address to match against the source MAC address of an outgoing IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv6 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VLAN. If you do not specify an interface ID, the appliance compares the PBR6 to

the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified interface. If you do not specify a value, the appliance compares the PBR6 to the outgoing packets on all interfaces.

priority

Priority of the PBR6, which determines the order in which it is evaluated relative to the other PBR6s. If you do not specify priorities while creating PBR6s, the PBR6s are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

state

Enable or disable the PBR6. After you apply the PBR6s, the NetScaler appliance compares outgoing packets to the enabled PBR6s.

Possible values: ENABLED, DISABLED

Default value: XACLENABLED

msr

Monitor the route specified by the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nextHop

IP address of the next hop router to which to send matching packets if action is set to ALLOW. This next hop should be directly reachable from the appliance.

nextHopVlan

VLAN number to be used for link local nexthop .

Minimum value: 1

Maximum value: 4094

Example

```
add ns pbr6 rule1 ALLOW -srcport 45-1024 -destIPv6 2001::45 -nexthop 2001::49
```

---

renumber ns pbr6

Renumbers the priorities of PBR6s to multiples of 10.To commit this operation, you must apply the PBR6s. Enables you to assign a new PBR6 a priority that is between two existing, consecutively numbered priorities. For example, if two PBR6s, PBR6-1 and PBR6-2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add PBR6-3 with priority 25.

Synopsys

```
renumber ns pbr6
```

Example

```
renumber pbr6
```

---

rm ns pbr6

Removes a PBR6 from the NetScaler appliance. To commit this operation, you must apply the PBR6s.

Synopsys

```
rm ns pbr6 <name> ...
```

Arguments

name

Name of the PBR6 that you want to remove.

Example

```
rm ns pbr6 rule1
```

---

set ns pbr6

Modifies the specified parameters of a PBR6.To commit this operation, you must apply the PBR6s.

Synopsys

set ns pbr6 <name> [-action (ALLOW | DENY)] [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr (ENABLED | DISABLED) [-monitor <string>]] [-nextHop <nextHopVal>] [-nextHopVlan <positive_integer>]

Arguments

name

Name of the PBR6 whose parameters you want to modify.

action

Action to perform on the outgoing IPv6 packets that match the PBR6.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

srcIPv6

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

srcPort

Source Port (range).

destIPv6

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destPort

Destination Port (range).

srcMac

MAC address to match against the source MAC address of an outgoing IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv6 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

interface

ID of an interface. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified interface. If you do not specify a value, the appliance compares the PBR6 to the outgoing packets on all interfaces.

priority

Priority of the PBR6, which determines the order in which it is evaluated relative to the other PBR6s. If you do not specify priorities while creating PBR6s, the PBR6s are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

msr

Monitor the route specified by the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nextHop

IP address of the next hop router to which to send matching packets if action is set to ALLOW. This next hop should be directly reachable from the appliance.

nextHopVlan

VLAN number to be used for link local nexthop .

Minimum value: 1

Maximum value: 4094

Example

```
set ns pbr6 rule1 -srcPort 50
```

---

unset ns pbr6

Resets the attributes of the specified PBR6. Attributes for which a default value is available revert to their default values. Refer to the set ns pbr6 command for descriptions of the parameters..Refer to the set ns pbr6 command for meanings of the arguments.

Synopsys

```
unset ns pbr6 <name> [-srcIPv6] [-srcPort] [-destIPv6] [-destPort] [-srcMac] [-protocol] [-interface] [-vlan] [-msr] [-monitor] [-nextHop] [-nextHopVlan]
```

Example

```
unset ns pbr6 rule1 -srcPort
```

---

enable ns pbr6

Enables a PBR6. To commit this operation, you must apply the PBR6s.After you apply the PBR6s, the NetScaler appliance compares outgoing packets to the enabled PBR6.

Synopsys

```
enable ns pbr6 <name> ...
```

Arguments

name

Name of PBR6 that you want to enable.

Example

```
enable ns pbr6 rule1
```

---

disable ns pbr6

Disables a PBR6. To commit this operation, you must apply the PBR6s.After you apply the PBR6s, the NetScaler appliance does not compare outgoing packets to the disabled PBR6s.

Synopsys

```
disable ns pbr6 <name> ...
```

Arguments

name

Name of PBR6 that you want to disable.

Example

```
disable ns pbr6 rule1
```

---

stat ns pbr6

Displays statistics related to the PBR6s. To display statistics of all the PBR6s, run the command without any parameters. To display statistics of a particular PBR6, specify the name of the PBR6.

Synopsys

```
stat ns pbr6 [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the PBR6 whose statistics you want the NetScaler appliance to display.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Allow PBR6 hits (PBR6Allow)

Total packets that matched the PBR6 with action ALLOW

Deny PBR6 hits (PBR6Deny)

Total packets that matched PBR6 with action DENY

PBR6 hits (PBR6TotHits)

Total packets that matched one of the configured PBR6

PBR6 misses (PBR6Miss)

Total packets that did not match any PBR6

Hits for this PBR6 (PBR6Hits)

Number of times the pbr6 was hit

Example

stat pbr6

---

show ns pbr6

Displays settings related to the PBR6s. To display settings of all the PBR6s, run the command without any parameters. To display settings of a particular PBR6, specify the name of the PBR6.

Synopsys

show ns pbr6 [<name>] [-detail]

Arguments

name

Name of the PBR6 whose settings you want the NetScaler appliance to display.

detail

To get a detailed view.

summary

fullValues

format

level

Outputs

td

Traffic Domain Id.

action

Action to perform on the outgoing IPv6 packets that match the PBR6.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

srcMac

MAC address to match against the source MAC address of an outgoing IPv6 packet.

stateflag

PBR6 state flag.

protocol

Protocol number in IPv6 header or name.

protocolNumber

Protocol number in IPv6 header or name.

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcIPv6Val

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destIPv6Val

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

vlan

ID of the VLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VLANs.

state

If this route is UP/DOWN.

kernelstate

Commit status of the PBR6.

interface

ID of an interface. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified interface. If you do not specify a value, the appliance compares the PBR6 to the outgoing packets on all interfaces.

hits

Number of hits of this PBR6.

priority

Priority of the PBR6, which determines the order in which it is evaluated relative to the other PBR6s. If you do not specify priorities while creating PBR6s, the PBR6s are evaluated in the order in which they are created.

operator

Logical operator.

msr

Whether Monitored Static Route(MSR) is enabled or disabled.

monitor

Name of the monitor, of type PING6 or ARP, configured on the NetScaler appliance to monitor the route specified by the Next Hop parameter. You must enable the MSR parameter before setting the Monitor parameter.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

nextHopVal

ID of the VLAN, if you have specified a link local address for the Next Hop parameter.

nextHopVlan

VLAN number to be used for link local nexthop .

data

Internal data of this route.

devno

count

Example

```
show ns pbr6 rule1 1)    Name: r1                Action: DENY      srcIPv6 = 2001::1      destIPv6      srcMac:
```

clear ns pbr6

Removes all PBR6s from the NetScaler appliance. This operation does not require an explicit apply.

Synopsys

```
clear ns pbr6
```

Example

```
clear ns pbr6
```

apply ns pbr6

Updates the PBR6's memory tree (lookup table), adding any new PBR6 and applying any modifications to the existing PBR6s. The lookup table includes the configuration of all the extended PBR6s on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the outgoing IPv6 packets.

Synopsys

```
apply ns pbr6
```

Example

```
apply ns pbr6
```

ns pbrs

Sep 22, 2015

The following operations can be performed on "ns pbrs":

renumber | clear | apply

renumber ns pbrs

Renumbers the priorities of PBRs to multiples of 10.To commit this operation, you must apply the PBRs. Enables you to assign a new PBR a priority that is between two existing, consecutively numbered priorities. For example, if two PBRs, PBR1 and PBR2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add PBR3 with priority 25.

Synopsys

renumber ns pbrs

Example

```
renumber pbrs
```

---

clear ns pbrs

Removes all PBRs from the NetScaler appliance. This operation does not require an explicit apply.

Synopsys

clear ns pbrs

Example

```
clear ns pbrs
```

---

apply ns pbrs

Updates the PBR's memory tree (lookup table), adding any new PBR and applying any modifications to existing PBRs. The lookup table includes the configuration of all the extended PBRs on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the outgoing IPv4 packets.

Synopsys

apply ns pbrs

Example

```
apply ns pbrs
```

---

ns persistencesession

Sep 22, 2015

The following operations can be performed on "ns persistencesession":

show | clear

show ns persistencesession

Get all Sessions corresponding to a Vserver NOTE: This command is deprecated.Moved to LB command group

Synopsys

Arguments

name

The name of the virtual server.

summary

fullValues

Outputs

type

The netmask of this IP.

srcIP

SOURCE IP.

srcIPv6

SOURCE IPv6 ADDRESS.

destIP

DESTINATION IP.

destIPv6

DESTINATION IPv6 ADDRESS.

flags

IPv6 FLAGS.

destPort

Destination port.

vServerName

Virtual server name.

timeout

Persistent Session timeout.

referenceCount

Reference Count.

sipCallID

SIP CALLID.NOTE: This attribute is deprecated.Replaced by "persistenceParam" field

persistenceParam

Specific persistence information . Callid in case of SIP_CALLID persistence entry , RTSP session id in case of RTSP_SESSIONID persistence entry.

devno

count

stateflag

Example

```
show ns persistencesession vipname
```

---

clear ns persistencesession

Use this command to clear/flush persistence sessions NOTE: This command is deprecated.Moved to LB command group

Synopsys

Arguments

vServer

The name of the LB vserver whose persistence sessions are to be flushed. If not specified, all persistence sessions will be flushed .

Example

```
clear persistenceSessions -vserver vip1
```

---

ns rateControl

Sep 22, 2015

The following operations can be performed on "ns rateControl":

set | unset | show

set ns rateControl

Sets the UDP/TCP/ICMP packet rate controls for any application that is not configured at System (direct access to the backend through System).

Synopsys

set ns rateControl [-tcpThreshold <positive_integer>] [-udpThreshold <positive_integer>] [-icmpThreshold <positive_integer>] [-tcprstThreshold <positive_integer>]

Arguments

tcpThreshold

Number of SYNs permitted per 10 milliseconds.

udpThreshold

Number of UDP packets permitted per 10 milliseconds.

icmpThreshold

Number of ICMP packets permitted per 10 milliseconds.

Default value: 100

tcprstThreshold

The number of TCP RST packets permitted per 10 milli second. zero means rate control is disabled and 0xffffffff means every thing is rate controlled

Default value: 100

Example

The following command will set the SYN rate to 100, icmp rate to 10 and the udp rate to unlimited. set ns ratecontrol -tcpThreshold 100 -udpThreshold 0 -icmpThreshol

---

unset ns rateControl

Use this command to remove ns rateControl settings.Refer to the set ns rateControl command for meanings of the arguments.

Synopsys

unset ns rateControl [-tcpThreshold] [-udpThreshold] [-icmpThreshold] [-tcprstThreshold]

---

show ns rateControl

Displays the values configured for rate control on the appliance.

Synopsys

show ns rateControl

Arguments

format

level

Outputs

tcpThreshold

Number of SYNs permitted per 10 milliseconds.

udpThreshold

Number of UDP packets permitted per 10 milliseconds.

icmpThreshold

Number of ICMP packets permitted per 10 milliseconds.

tcprstThreshold

The number of TCP RST packets permitted per 10 milli second. zero means rate control is disabled and 0xffffffff means every thing is rate controlled

Example

By default, there is no rate control for TCP/UDP and for ICMP it will be 100. The output of the "show ns ratecontrol" command, with default setting, > show ns ra

ns rollbackcmd

Sep 22, 2015

The following operations can be performed on "ns rollbackcmd":

show ns rollbackcmd

Generates the command(s) that can be used to roll back the command(s) that are specified in an input file. For example, if you want to roll back the creation of a load balancing virtual server named vserver_test, you must include the 'add lb vserver vserver_test ..' command in the input file. The output of this command is the 'rm lb vserver vserver_test' command.

Synopsys

show ns rollbackcmd [-fileName <input_filename>] [-outtype (cli | xml)]

Arguments

fileName

File that contains the commands for which the rollback commands must be generated. Specify the full path of the file name.

outtype

Format in which the rollback commands must be generated.

Possible values: cli, xml

Example

```
show ns rollbackcmd -file <file_name>
```

ns rpcNode

Sep 22, 2015

The following operations can be performed on "ns rpcNode":

set | unset | show

set ns rpcNode

Sets the authentication attributes associated with peer system node. All system nodes use Remote Procedure Calls (RPC) to communicate.

Synopsys

set ns rpcNode <IPAddress> {-password } [-srcIP <ip_addr|ipv6_addr|*>] [-secure (YES | NO)]

Arguments

IPAddress

IP address of the node. This has to be in the same subnet as the NSIP address.

password

Password to be used in authentication with the peer system node.

srcIP

Source IP address to be used to communicate with the peer system node. The default value is 0, which means that the appliance uses the NSIP address as the source IP address.

secure

State of the channel when talking to the node.

Possible values: YES, NO

Example

Example-1: Failover configuration In a failover configuration define peer NS as: add node 1 10.101.4.87 Set peer ha-unit's password as: set ns rpcnode 10.101.4.87 -p

---

unset ns rpcNode

Use this command to remove ns rpcNode settings.Refer to the set ns rpcNode command for meanings of the arguments.

Synopsys

unset ns rpcNode <IPAddress> [-password] [-srcIP] [-secure]

---

show ns rpcNode

Display a list of nodes currently communicating by using Remote Procedure Calls (RPC).

Synopsys

show ns rpcNode [<IPAddress>]

Arguments

IPAddress

IP address of the node.

summary

fullValues

format

level

Outputs

password

Password.

srcIP

The src ip used in communiction with the peer System node.

secure

State of the channel when talking to the node.

flags

Flags related to this rpcNode

devno

count

stateflag

Example

Following example shows list of nodes communicating using RPC: > sh rpcnode 1) IPAddress: 10.101.4.84 Password: ..8a7b474124

ns runningConfig

Sep 22, 2015

The following operations can be performed on "ns runningConfig":

show ns runningConfig

Displays all the configurations that have been executed on the appliance, including the configurations that have not yet been saved. Note: The unsaved configurations are lost when the appliance is rebooted or shut down.

Synopsys

show ns runningConfig [-withDefaults]

Arguments

withDefaults

Include default values of parameters that have not been explicitly configured. If this argument is disabled, such parameters are not included.

Outputs

response

running config data as text blob

ns savedConfig

Sep 22, 2015

The following operations can be performed on "ns savedConfig":

show ns savedConfig

Displays the saved configurations.

Synopsys

show ns savedConfig

Outputs

textBlob

Text of the last saved configuration.

ns simpleacl

Sep 22, 2015

The following operations can be performed on "ns simpleacl":

add | clear | rm | flush | show | stat

add ns simpleacl

Adds a simple ACL rule to the NetScaler appliance. Simple ACL rules filter IPv4 packets on the basis of their source IP addresses and, optionally, the destination port and/or protocol. Any packet with the characteristics specified in the simple ACL rule is dropped.

Synopsys

add ns simpleacl <aclname> <aclaction> [-td <positive_integer>] -srcIP <ip_addr> [-destPort <port> -protocol (TCP | UDP)] [-TTL <positive_integer>]

Arguments

aclname

Name for the simple ACL rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the simple ACL rule is created.

aclaction

Drop incoming IPv4 packets that match the simple ACL rule.

Possible values: DENY

td

Traffic Domain Id.

Maximum value: 4094

srcIP

IP address to match against the source IP address of an incoming IPv4 packet.

destPort

Port number to match against the destination port number of an incoming IPv4 packet.

Omitting the port number creates an all-ports simple ACL rule, which matches any port. In that case, you cannot create another simple ACL rule specifying a specific port and the same source IPv4 address.

TTL

Number of seconds, in multiples of four, after which the simple ACL rule expires. If you do not want the simple ACL rule to expire, do not specify a TTL value.

Minimum value: 4

Maximum value: 2147483647

Example

```
add simpleacl rule1 DENY -srcIP 1.1.1.1 -destPort 80 -protocol TCP add simpleacl rule2 DENY -srcIP 2.2.2.2 -TTL 600
```

---

clear ns simpleacl

Removes all simple ACL rules from the NetScaler appliance.

Synopsys

```
clear ns simpleacl
```

---

rm ns simpleacl

Removes a simple ACL rule from the NetScaler appliance.

Synopsys

rm ns simpleacl <aclname> ...

Arguments

aclname

Name of the simple ACL rule that you want to remove.

Example

```
rm ns simpleacl rule1
```

flush ns simpleacl

Terminates all established IPv4 connections that match any of the newly configured simple ACL rules. Note: If you plan to create more than one simple ACL rule and flush existing connections that match any of them, you can minimize the affect on performance by first creating all of the simple ACL rules and then running flush only once.

Synopsys

flush ns simpleacl -estSessions

Arguments

estSessions

show ns simpleacl

Displays settings of all the simple ACL rules or of the specified simple ACL rule. To display settings of all the simple ACL rules, run the command without any parameters. To display settings of a particular simple ACL rule, specify the name of the simple ACL rule.

Synopsys

show ns simpleacl [<aclname>]

Arguments

aclname

Name of the simple ACL rule whose details you want the NetScaler appliance to display.

summary

fullValues

format

level

Outputs

aclaction

Drop incoming IPv4 packets that match the simple ACL rule.

td

Traffic Domain Id.

srcIP

Source IP address.

stateflag

SACL state flag.

destPort

Destination Port.

protocol

Protocol associated with the ACL rule.

TTL

Time to expire this ACL rule(in seconds).

time

Time when this acl is added.

hits

Number of hits for this ACL rule.

devno

count

Example

```
show simpleacl rule1  Name: rule1                Action: DENY  srcIP = 10.102.1.150  Protocol = TCP                DestPort = 110  Hits:
```

stat ns simpleacl

Displays statistics related to the simple ACL rules.

Synopsys

stat ns simpleacl [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

SimpleACL hits (SACLHits)

Packets matching a SimpleACL.

SimpleACL misses (SACLMiss)

Packets not matching any SimpleACL.

SimpleACLs count (SACLsCount)

Number of SimpleACLs configured.

Allow SimpleACL hits (SACLAllow)

Total packets that matched a SimpleACL with action ALLOW and got consumed by NetScaler.

Bridge SimpleACL hits (SACLBdg)

Total packets that matched a SimpleACL with action BRIDGE and got bridged by NetScaler.

Deny SimpleACL hits (SACLDeny)

Packets dropped because they match SimpleACL (Access Control List) with processing mode set to DENY.

Example

stat simpleacl

ns simpleacl6

Sep 22, 2015

The following operations can be performed on "ns simpleacl6":

add | clear | flush | rm | show | stat

add ns simpleacl6

Adds a simple ACL6 rule to the NetScaler appliance. Simple ACL6 rules filter IPv6 packets on the basis of their source IP addresses and, optionally, the destination port and/or protocol. Any packet with the characteristics specified in the simple ACL6 rule is dropped.

Synopsys

add ns simpleacl6 <aclname> [-td <positive_integer>] <aclaction> -srcIPv6 <ipv6_addr|null> [-destPort <port> -protocol (TCP | UDP)] [-TTL <positive_integer>]

Arguments

aclname

Name for the simple ACL6 rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the simple ACL6 rule is created.

td

Traffic Domain Id.

Maximum value: 4094

aclaction

Drop incoming IPv6 packets that match the simple ACL6 rule.

Possible values: DENY

srcIPv6

IP address to match against the source IP address of an incoming IPv6 packet.

destPort

Port number to match against the destination port number of an incoming IPv6 packet.

Omitting the port number creates an all-ports simple ACL6 rule, which matches any port. In that case, you cannot create another simple ACL6 rule specifying a specific port and the same source IPv6 address.

TTL

Number of seconds, in multiples of four, after which the simple ACL6 rule expires. If you do not want the simple ACL6 rule to expire, do not specify a TTL value.

Minimum value: 4

Maximum value: 2147483647

Example

```
add simpleacl6 rule1 DENY -srcIP6 fe80::2c0:95ff:fec5:d9b8 -destPort 80 -protocol TCP add simpleacl rule2 DENY -srcIP6 3ffe:100:100::1 -TTL 600
```

---

clear ns simpleacl6

Removes all simple ACL6 rules from the NetScaler appliance.

Synopsys

clear ns simpleacl6

Example

```
clear ns simpleacl6
```

---

flush ns simpleacl6

Terminates all established IPv6 connections that match any of the newly configured simple ACL6 rules. Note: If you plan to create more than one simple ACL6 rule and flush existing connections that match any of them, you can minimize the affect on performance by first creating all of the simple ACL6 rules and then running flush only once.

Synopsys

flush ns simpleacl6 -estSessions

Arguments

estSessions

---

rm ns simpleacl6

Removes a simple ACL6 rule from the NetScaler appliance.

Synopsys

rm ns simpleacl6 <aclname> ...

Arguments

aclname

Name of the simple ACL6 rule that you want to remove.

Example

```
rm ns simpleacl6 rule1
```

---

show ns simpleacl6

Displays settings of all the simple ACL6 rules or of the specified simple ACL6 rule. To display settings of all the simple ACL6 rules, run the command without any parameters. To display settings of a particular simple ACL6 rule, specify the name of the simple ACL6 rule.

Synopsys

show ns simpleacl6 [<aclname>]

Arguments

aclname

Name of the simple ACL6 rule whose settings you want the NetScaler appliance to display.

summary

fullValues

format

level

Outputs

aclaction

Action associated with the SACL6 rule.

td

Traffic Domain Id.

srcIPv6

Source IP6 address.

stateflag

SACL6 state flag.

destPort

Destination Port.

protocol

Protocol associated with the ACL rule.

TTL

Time to expire this ACL rule(in seconds).

hits

Number of hits for this SACL6 rule.

time

Time when this acl is added.

devno

count

Example

```
show simpleacl6 rule1  Name: rule1    Action: DENY              Hits: 5  srcIP6 = 3ffe:100:100::1  Protocol = TCP             DestPort
```

---

stat ns simpleacl6

Displays statistics related to the simple ACL6 rules.

Synopsys

stat ns simpleacl6 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

SimpleACL6 hits (SACL6Hits)

Packets matching a SimpleACL6.

SimpleACL6 misses (SACL6Miss)

Packets not matching any SimpleACL6.

SimpleACLs6 count (SACL6sCount)

Number of SimpleACL6s configured.

Allow SimpleACL6 hits (SACL6Allow)

Total packets that matched a SimpleACL6 with action ALLOW and got consumed by NetScaler.

Bridge SimpleACL6 hits (SACL6Bdg)

Total packets that matched a SimpleACL6 with action BRIDGE and got bridged by NetScaler.

Deny SimpleACL6 hits (SACL6Deny)

Packets dropped because they match SimpleACL6 with processing mode set to DENY.

Example

stat simpleacl6

ns spParams

Sep 22, 2015

The following operations can be performed on "ns spParams":

set | unset | show

set ns spParams

Sets surge protection attributes on the appliance.

Synopsys

set ns spParams [-baseThreshold <integer>] [-throttle <throttle>]

Arguments

baseThreshold

Maximum number of server connections that can be opened before surge protection is activated.

Default value: 200

Maximum value: 32767

throttle

Rate at which the system opens connections to the server.

Possible values: Aggressive, Normal, Relaxed

Default value: NORM_SP_TABLE

Example

```
set ns spparams -baseThreshold 1000 -throttle aggressive set ns spparams -throttle relaxed
```

---

unset ns spParams

Use this command to remove ns spParams settings.Refer to the set ns spParams command for meanings of the arguments.

Synopsys

```
unset ns spParams [-baseThreshold] [-throttle]
```

---

show ns spParams

Displays the surge protection configuration on the appliance. Surge protection parameters are set by using the 'set ns spParams' command.

Synopsys

show ns spParams

Arguments

format

level

Outputs

baseThreshold

The base threshold. This is the maximum number of server connections that can be open before surge protection is activated.

throttle

Rate at which the system opens connections to the server.

Table

Table.

Example

```
> show ns spparams   Surge Protection parameters:   BaseThreshold: 200   Throttle:  Normal   Done
```

ns stats

Sep 22, 2015

The following operations can be performed on "ns stats":

show | clear

show ns stats

show ns stats is an alias for stat ns

Synopsys

show ns stats - alias for 'stat ns'

---

clear ns stats

Clearing stats

Synopsys

clear ns stats <cleanuplevel>

Arguments

cleanuplevel

The level of stats to be cleared. 'global' option will clear global counters only, 'all' option will clear all device counters also along with global counters. For both the cases only 'ever incrementing counters' i.e. total counters will be cleared.

Possible values: global, all

---

ns surgeQ

Sep 22, 2015

The following operations can be performed on "ns surgeQ":

flush ns surgeQ

Flushes the connections that are waiting in SurgeQ. SurgeQ contains the client connections waiting for a server connection.

Synopsys

flush ns surgeQ [-name <string> [-serverName <string> <port>]]

Arguments

name

Name of a virtual server, service or service group for which the SurgeQ must be flushed.

serverName

Name of a service group member. This argument is needed when you want to flush the SurgeQ of a service group.

Example

To flush the surgeQ system wide, use the command: flush ns SurgeQ. To flush the surgeQ specific to a vserver/service/svcgrp use the command: flush ns SurgeQ -name <n

ns tcpParam

Sep 22, 2015

The following operations can be performed on "ns tcpParam":

set | unset | show

set ns tcpParam

Sets the TCP parameters for the NetScaler appliance.

Synopsys

set ns tcpParam [-WS (ENABLED | DISABLED)] [-WSVal <positive_integer>] [-SACK (ENABLED | DISABLED)] [-learnVsvrMSS (ENABLED | DISABLED)] [-maxBurst <positive_integer>] [-initialCwnd <positive_integer>] [-delayedAck <positive_integer>] [-downStateRST (ENABLED | DISABLED)] [-nagle (ENABLED | DISABLED)] [-limitedPersist (ENABLED | DISABLED)] [-oooQSize <positive_integer>] [-ackOnPush (ENABLED | DISABLED)] [-maxPktPerMss <integer>] [-pktPerRetx <integer>] [-minRTO <integer>] [-slowStartIncr <integer>] [-maxDynServerProbes <positive_integer>] [-synHoldFastGiveup <positive_integer>] [-maxSynholdPerprobe <positive_integer>] [-maxSynhold <positive_integer>] [-mssLearnInterval <positive_integer>] [-mssLearnDelay <positive_integer>] [-maxTimeWaitConn <positive_integer>] [-maxSynAckRetx <positive_integer>] [-synAttackDetection (ENABLED | DISABLED)] [-connFlushIfNoMem <connFlushIfNoMem>] [-connFlushThres <positive_integer>] [-mptcpConCloseOnPassiveSF (ENABLED | DISABLED)] [-mptcpChecksum (ENABLED | DISABLED)] [-mptcpSFtimeout <secs>] [-mptcpSFReplaceTimeout <secs>] [-mptcpMaxSF <positive_integer>] [-mptcpMaxPendingSF <positive_integer>] [-mptcpPendingJoinThreshold <positive_integer>] [-mptcpRTOsToSwitchSF <positive_integer>] [-mptcpUseBackupOnDSS (ENABLED | DISABLED)] [-TcpMaxRetries <positive_integer>] [-mptcpImmediateSFCloseOnFIN (ENABLED | DISABLED)]

Arguments

WS

Enable or disable window scaling.

Possible values: ENABLED, DISABLED

Default value: DISABLED

WSVal

Factor used to calculate the new window size.

This argument is needed only when the window scaling is enabled.

Default value: 4

Maximum value: 14

SACK

Enable or disable Selective ACKnowledgement (SACK).

Possible values: ENABLED, DISABLED

Default value: DISABLED

learnVsvrMSS

Enable or disable maximum segment size (MSS) learning for virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxBurst

Maximum number of TCP segments allowed in a burst.

Default value: 6

Minimum value: 1

Maximum value: 255

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

Default value: 4

Minimum value: 1

Maximum value: 44

recvBuffSize

TCP Receive buffer size

Default value: 8190

Minimum value: 8190

Maximum value: 20971520

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

Default value: 100

Minimum value: 10

Maximum value: 300

downStateRST

Flag to switch on RST on down services.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nagle

Enable or disable the Nagle algorithm on TCP connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

limitedPersist

Limit the number of persist (zero window) probes.

Possible values: ENABLED, DISABLED

Default value: ENABLED

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means infinite.

Default value: 64

Maximum value: 65535

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxPktPerMss

Maximum number of TCP packets allowed per maximum segment size (MSS).

Maximum value: 1460

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

Default value: 1

Minimum value: 1

Maximum value: 100

minRTO

Minimum retransmission timeout, in milliseconds, specified in 10-millisecond increments (value must yield a whole number if divided by 10).

Default value: 1000

Minimum value: 10

Maximum value: 64000

slowStartIncr

Multiplier that determines the rate at which slow start increases the size of the TCP transmission window after each acknowledgement of successful transmission.

Default value: 2

Minimum value: 1

Maximum value: 100

maxDynServerProbes

Maximum number of probes that NetScaler can send out in 10 milliseconds, to dynamically learn a service. NetScaler probes for the existence of the origin in case of wildcard virtual server or services.

Default value: 7

Minimum value: 1

Maximum value: 65535

synHoldFastGiveup

Maximum threshold. After crossing this threshold number of outstanding probes for origin, the NetScaler reduces the number of connection retries for probe connections.

Default value: 1024

Minimum value: 256

Maximum value: 65535

maxSynholdPerprobe

Limit the number of client connections (SYN) waiting for status of single probe. Any new SYN packets will be dropped.

Default value: 128

Minimum value: 1

Maximum value: 255

maxSynhold

Limit the number of client connections (SYN) waiting for status of probe system wide. Any new SYN packets will be dropped.

Default value: 16384

Minimum value: 256

Maximum value: 65535

mssLearnInterval

Duration, in seconds, to sample the Maximum Segment Size (MSS) of the services. The NetScaler appliance determines the best MSS to set for the virtual server based on this sampling. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

Default value: 180

Minimum value: 1

Maximum value: 1048576

mssLearnDelay

Frequency, in seconds, at which the virtual servers learn the Maximum segment size (MSS) from the services. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

Default value: 3600

Minimum value: 1

Maximum value: 1048576

maxTimeWaitConn

Maximum number of connections to hold in the TCP TIME_WAIT state. New connections entering TIME_WAIT state are proactively cleaned up.

Default value: 7000

Minimum value: 1

KAprobeUpdateLastactivity

Update last activity for KA probes

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxSynAckRetx

When 'syncookie' is disabled in the TCP profile that is bound to the virtual server or service, and the number of TCP SYN+ACK retransmission by NetScaler for that virtual server or service crosses this threshold, the NetScaler appliance responds by using the TCP SYN-Cookie mechanism.

Default value: 100

Minimum value: 100

Maximum value: 1048576

synAttackDetection

Detect TCP SYN packet flood and send an SNMP trap.

Possible values: ENABLED, DISABLED

Default value: ENABLED

connFlushIfNoMem

Flush an existing connection if no memory can be obtained for new connection.

HALF_CLOSED_AND_IDLE: Flush a connection that is closed by us but not by peer, or failing that, a connection that is past configured idle time. New connection fails if no such connection can be found.

FIFO: If no half-closed or idle connection can be found, flush the oldest non-management connection, even if it is active. New connection fails if the oldest few connections are management connections.

Note: If you enable this setting, you should also consider lowering the zombie timeout and half-close timeout (see NSCLI command: set ns timeout).

See Also: connFlushThres argument below.

Possible values: NONE, HALFCLOSED_AND_IDLE, FIFO

Default value: NSA_CONNFLUSH_NONE

connFlushThres

Flush an existing connection (as configured through -connFlushIfNoMem FIFO) if the system has more than specified number of connections, and a new connection is to be established. Note: This value may be rounded down to be a whole multiple of the number of packet engines running.

Minimum value: 1

mptcpConCloseOnPassiveSF

Accept DATA_FIN/FAST_CLOSE on passive subflow

Possible values: ENABLED, DISABLED

Default value: ENABLED

mptcpChecksum

Use MPTCP DSS checksum

Possible values: ENABLED, DISABLED

Default value: ENABLED

mptcpSFtimeout

The timeout value in seconds for idle mptcp subflows. If this timeout is not set, idle subflows are cleared after cltTimeout of vserver

Maximum value: 31536000

mptcpSFReplaceTimeout

The minimum idle time value in seconds for idle mptcp subflows after which the sublow is replaced by new incoming subflow if maximum subflow limit is reached. The priority for replacement is given to those subflow without any transaction

Default value: 10

Maximum value: 31536000

mptcpMaxSF

Maximum number of subflow connections supported in established state per mptcp connection.

Default value: 4

Minimum value: 2

Maximum value: 6

mptcpMaxPendingSF

Maximum number of subflow connections supported in pending join state per mptcp connection.

Default value: 4

Maximum value: 4

mptcpPendingJoinThreshold

Maximum system level pending join connections allowed.

Maximum value: 4294967294

mptcpRTOsToSwitchSF

Number of RTO's at subflow level, after which MPCTP should start using other subflow.

Default value: 2

Minimum value: 1

Maximum value: 6

mptcpUseBackupOnDSS

When enabled, if NS receives a DSS on a backup subflow, NS will start using that subflow to send data. And if disabled, NS will continue to transmit on current chosen subflow. In case there is some error on a subflow (like RTO's/RST etc.) then NS can choose a backup subflow irrespective of this tunable.

Possible values: ENABLED, DISABLED

Default value: ENABLED

TcpMaxRetries

Number of RTO's after which a connection should be freed.

Default value: 7

Minimum value: 1

Maximum value: 7

mptcpImmediateSFCloseOnFIN

Allow subflows to close immediately on FIN before the DATA_FIN exchange is completed at mptcp level.

Possible values: ENABLED, DISABLED

Default value: DISABLED

---

unset ns tcpParam

Use this command to remove ns tcpParam settings.Refer to the set ns tcpParam command for meanings of the arguments.

Synopsys

unset ns tcpParam [-WS] [-WSVal] [-SACK] [-learnVsvrMSS] [-maxBurst] [-initialCwnd] [-delayedAck] [-downStateRST] [-nagle] [-limitedPersist] [-oooQSize] [-ackOnPush] [-maxPktPerMss] [-pktPerRetx] [-minRTO] [-slowStartIncr] [-maxDynServerProbes] [-synHoldFastGiveup] [-maxSynholdPerprobe] [-maxSynhold] [-mssLearnInterval] [-mssLearnDelay] [-maxTimeWaitConn] [-maxSynAckRetx] [-synAttackDetection] [-connFlushIfNoMem] [-connFlushThres] [-mptcpConCloseOnPassiveSF] [-mptcpChecksum] [-mptcpSFtimeout] [-mptcpSFReplaceTimeout] [-mptcpMaxSF] [-mptcpMaxPendingSF] [-mptcpPendingJoinThreshold] [-mptcpRTOsToSwitchSF] [-mptcpUseBackupOnDSS] [-TcpMaxRetries] [-mptcpImmediateSFCloseOnFIN]

---

show ns tcpParam

Displays the TCP parameters configured on the NetScaler appliance.

Synopsys

show ns tcpParam

Arguments

format

level

Outputs

WS

Enable or disable window scaling.

WSVal

Factor used to calculate the new window size.

This argument is needed only when the window scaling is enabled.

SACK

Enable or disable Selective ACKnowledgement (SACK).

learnVsvrMSS

Enable or disable maximum segment size (MSS) learning for virtual servers.

maxBurst

Maximum number of TCP segments allowed in a burst.

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

recvBuffSize

TCP Receive buffer sizeNOTE: This attribute is deprecated.This option is deprecated in favour of -buffersize

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

downStateRST

Flag to switch on RST on down services.

nagle

Enable or disable the Nagle algorithm on TCP connections.

limitedPersist

Limit the number of persist (zero window) probes.

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means infinite.

ackOnPush

Immediate ACK on PUSH packet

maxPktPerMss

Maximum packets per MSS

pktPerRetx

Maximum packets per retransmission

minRTO

Minimum retransmission timeout, in milliseconds, specified in 10-millisecond increments (value must yield a whole number if divided by 10).

slowStartIncr

TCP slowstart increment factor

maxDynServerProbes

Maximum number of probes that NetScaler can send out in 10 milliseconds, to dynamically learn a service. NetScaler probes for the existence of the origin in case of wildcard virtual server or services.

synHoldFastGiveup

Maximum threshold. After crossing this threshold number of outstanding probes for origin, the NetScaler reduces the number of connection retries for probe connections.

maxSynholdPerprobe

Limit the number of client connections (SYN) waiting for status of single probe. Any new SYN packets will be dropped.

maxSynhold

Limit the number of client connections (SYN) waiting for status of probe system wide. Any new SYN packets will be dropped.

mssLearnInterval

Duration, in seconds, to sample the Maximum Segment Size (MSS) of the services. The NetScaler appliance determines the best MSS to set for the virtual server based on this sampling. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

mssLearnDelay

Frequency, in seconds, at which the virtual servers learn the Maximum segment size (MSS) from the services. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

maxTimeWaitConn

Maximum number of connections to hold in the TCP TIME_WAIT state. New connections entering TIME_WAIT state are proactively cleaned up.

KAprobeUpdateLastactivity

Update last activity for KA probesNOTE: This attribute is deprecated.This option has been moved tcpProfile

maxSynAckRetx

When 'syncookie' is disabled in the TCP profile that is bound to the virtual server or service, and the number of TCP SYN+ACK retransmission by NetScaler for that virtual server or service crosses this threshold, the NetScaler appliance responds by using the TCP SYN-Cookie mechanism.

synAttackDetection

Detect TCP SYN packet flood and send an SNMP trap.

connFlushIfNoMem

Flush an existing connection if no memory can be obtained for new connection.

HALF_CLOSED_AND_IDLE: Flush a connection that is closed by us but not by peer, or failing that, a connection that is past configured idle time. New connection fails if no such connection can be found.

FIFO: If no half-closed or idle connection can be found, flush the oldest non-management connection, even if it is active. New connection fails if the oldest few connections are management connections.

Note: If you enable this setting, you should also consider lowering the zombie timeout and half-close timeout (see NSCLI command: set ns timeout).

See Also: connFlushThres argument below.

connFlushThres

Flush an existing connection (as configured through -connFlushIfNoMem FIFO) if the system has more than specified number of connections, and a new connection is to be established. Note: This value may be rounded down to be a whole multiple of the number of packet engines running.

mptcpConCloseOnPassiveSF

Accept DATA_FIN/FAST_CLOSE on passive subflow

mptcpChecksum

Use MPTCP DSS checksum

mptcpSFtimeout

The timeout value in seconds for idle mptcp subflows. If this timeout is not set, idle subflows are cleared after cltTimeout of vserver

mptcpSFReplaceTimeout

The minimum idle time value in seconds for idle mptcp subflows after which the sublow is replaced by new incoming subflow if maximum subflow limit is reached. The priority for replacement is given to those subflow without any transaction

mptcpMaxSF

Maximum number of subflow connections supported in established state per mptcp connection.

mptcpMaxPendingSF

Maximum number of subflow connections supported in pending join state per mptcp connection.

mptcpPendingJoinThreshold

Maximum system level pending join connections allowed.

mptcpRTOsToSwitchSF

Number of RTO's at subflow level, after which MPCTP should start using other subflow.

mptcpUseBackupOnDSS

When enabled, if NS receives a DSS on a backup subflow, NS will start using that subflow to send data. And if disabled, NS will continue to transmit on current chosen subflow. In case there is some error on a subflow (like RTO's/RST etc.) then NS can choose a backup subflow irrespective of this tunable.

TcpMaxRetries

Number of RTO's after which a connection should be freed.

mptcpImmediateSFCloseOnFIN

Allow subflows to close immediately on FIN before the DATA_FIN exchange is completed at mptcp level.

ns tcpProfile

Sep 22, 2015

The following operations can be performed on "ns tcpProfile":

add | rm | set | unset | show

add ns tcpProfile

Adds a TCP profile to the NetScaler appliance.

Synopsys

add ns tcpProfile <name> [-WS (ENABLED | DISABLED)] [-SACK (ENABLED | DISABLED)] [-WSVal <positive_integer>] [-nagle (ENABLED | DISABLED)] [-ackOnPush (ENABLED | DISABLED)] [-mss <positive_integer>] [-maxBurst <positive_integer>] [-initialCwnd <positive_integer>] [-delayedAck <positive_integer>] [-oooQSize <positive_integer>] [-maxPktPerMss <positive_integer>] [-pktPerRetx <positive_integer>] [-minRTO <positive_integer>] [-slowStartIncr <positive_integer>] [-bufferSize <positive_integer>] [-synCookie (ENABLED | DISABLED)] [-KAprobeUpdateLastactivity (ENABLED | DISABLED)] [-flavor (Default | Westwood)] [-dynamicReceiveBuffering (ENABLED | DISABLED)] [-KA (ENABLED | DISABLED)] [-KAconnIdleTime <positive_integer>] [-KAmaxProbes <positive_integer>] [-KAprobeInterval <positive_integer>] [-sendBuffsize <positive_integer>] [-mptcp (ENABLED | DISABLED)] [-EstablishClientConn <EstablishClientConn>]

Arguments

name

Name for a TCP profile. Must begin with a letter, number, or the underscore \\(_\\) character. Other characters allowed, after the first character, are the hyphen \\(-\\), period \\(.\\), hash \\(\\#\\), space \\(\\), at \\(@\\), and equal \\(=\\) characters. The name of a TCP profile cannot be changed after it is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my tcp profile" or 'my tcp profile'\\).

WS

Enable or disable window scaling.

Possible values: ENABLED, DISABLED

Default value: DISABLED

SACK

Enable or disable Selective ACKnowledgement (SACK).

Possible values: ENABLED, DISABLED

Default value: DISABLED

WSVal

Factor used to calculate the new window size.

This argument is needed only when window scaling is enabled.

Default value: 4

Maximum value: 14

nagle

Enable or disable the Nagle algorithm on TCP connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mss

Maximum number of octets to allow in a TCP data segment.

Maximum value: 1460

maxBurst

Maximum number of TCP segments allowed in a burst.

Default value: 6

Minimum value: 1

Maximum value: 255

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

Default value: 4

Minimum value: 1

Maximum value: 44

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

Default value: 100

Minimum value: 10

Maximum value: 300

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means infinite.

Default value: 64

Maximum value: 65535

maxPktPerMss

Maximum number of TCP packets allowed per maximum segment size (MSS).

Maximum value: 1460

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

Default value: 1

Minimum value: 1

Maximum value: 512

minRTO

Minimum retransmission timeout, in milliseconds.

Default value: 1000

Minimum value: 10

Maximum value: 64000

slowStartIncr

Multiplier that determines the rate at which slow start increases the size of the TCP transmission window after each acknowledgement of successful transmission.

Default value: 2

Minimum value: 1

Maximum value: 100

bufferSize

TCP buffering size, in bytes.

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

synCookie

Enable or disable the SYNCOOKIE mechanism for TCP handshake with clients. Disabling SYNCOOKIE prevents SYN attack protection on the NetScaler appliance.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KAprobeUpdateLastactivity

Update last activity for the connection after receiving keep-alive (KA) probes.

Possible values: ENABLED, DISABLED

Default value: ENABLED

flavor

Set TCP congestion control algorithm.

Possible values: Default, Westwood

Default value: NS_TCP_DEFAULT

dynamicReceiveBuffering

Enable or disable dynamic receive buffering. When enabled, allows the receive buffer to be adjusted dynamically based on memory and network conditions.

Note: The buffer size argument must be set for dynamic adjustments to take place.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KA

Send periodic TCP keep-alive (KA) probes to check if peer is still up.

Possible values: ENABLED, DISABLED

Default value: DISABLED

KAconnIdleTime

Duration, in seconds, for the connection to be idle, before sending a keep-alive (KA) probe.

Default value: NSTCP_KA_DEFAULT_CONN_IDLETIME

Minimum value: 1

Maximum value: 4095

KAmaxProbes

Number of keep-alive (KA) probes to be sent when not acknowledged, before assuming the peer to be down.

Default value: NSTCP_KA_DEFAULT_PROBE_COUNT

Minimum value: 1

Maximum value: 255

KAprobeInterval

Time interval, in seconds, before the next keep-alive (KA) probe, if the peer does not respond.

Default value: NSTCP_KA_DEFAULT_INTERVAL

Minimum value: 1

Maximum value: 4095

sendBuffsize

TCP Send Buffer Size

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

mptcp

Enable/Disable Multi-Path TCP

Possible values: ENABLED, DISABLED

Default value: DISABLED

EstablishClientConn

Establishing Client Client connection on First data/ Final-ACK / Automatic

Possible values: AUTOMATIC, CONN_ESTABLISHED, ON_FIRST_DATA

Default value: NS_CONN_AUTOMATIC

Example

```
add tcpprofile <profile name> -WS ENABLED -WSVAL 4
```

---

rm ns tcpProfile

Removes a TCP profile from the appliance.

Synopsys

rm ns tcpProfile <name>

Arguments

name

Name of the TCP profile to be removed.

Example

rm tcpprofile <profile name>

---

set ns tcpProfile

Modifies the attributes of a TCP profile.

Synopsys

set ns tcpProfile <name> [-WS (ENABLED | DISABLED)] [-SACK (ENABLED | DISABLED)] [-WSVal <positive_integer>] [-nagle (ENABLED | DISABLED)] [-ackOnPush (ENABLED | DISABLED)] [-mss <positive_integer>] [-maxBurst <positive_integer>] [-initialCwnd <positive_integer>] [-delayedAck <positive_integer>] [-oooQSize <integer>] [-maxPktPerMss <positive_integer>] [-pktPerRetx <positive_integer>] [-minRTO <positive_integer>] [-slowStartIncr <positive_integer>] [-bufferSize <positive_integer>] [-synCookie (ENABLED | DISABLED)] [-KAprobeUpdateLastactivity (ENABLED | DISABLED)] [-flavor (Default | Westwood)] [-dynamicReceiveBuffering (ENABLED | DISABLED)] [-KA (ENABLED | DISABLED)] [-KAconnIdleTime <positive_integer>] [-KAmaxProbes <positive_integer>] [-KAprobeInterval <positive_integer>] [-sendBuffsize <positive_integer>] [-mptcp (ENABLED | DISABLED)] [-EstablishClientConn <EstablishClientConn>]

Arguments

name

Name of the TCP profile to be modified.

WS

Enable or disable window scaling.

Possible values: ENABLED, DISABLED

Default value: DISABLED

SACK

Enable or disable Selective ACKnowledgement (SACK).

Possible values: ENABLED, DISABLED

Default value: DISABLED

WSVal

Factor used to calculate the new window size.

This argument is needed only when window scaling is enabled.

Default value: 4

Maximum value: 14

nagle

Enable or disable the Nagle algorithm on TCP connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mss

Set Maximum Segment Size(MSS) to use for TCP Connection(0 forces use of global setting)

Maximum value: 1460

maxBurst

Maximum number of TCP segments allowed in a burst.

Default value: 6

Minimum value: 1

Maximum value: 255

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

Default value: 4

Minimum value: 1

Maximum value: 44

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

Default value: 100

Minimum value: 10

Maximum value: 300

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means infinite.

Default value: 64

Maximum value: 65535

maxPktPerMss

Maximum number of TCP packets allowed per maximum segment size (MSS).

Maximum value: 1460

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

Default value: 1

Minimum value: 1

Maximum value: 512

minRTO

Minimum retransmission timeout, in milliseconds.

Default value: 1000

Minimum value: 10

Maximum value: 64000

slowStartIncr

Multiplier that determines the rate at which slow start increases the size of the TCP transmission window after each acknowledgement of successful transmission.

Default value: 2

Minimum value: 1

Maximum value: 100

bufferSize

TCP buffering size, in bytes.

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

synCookie

Enable or disable the SYNCOOKIE mechanism for TCP handshake with clients. Disabling SYNCOOKIE prevents SYN attack protection on the NetScaler appliance.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KAprobeUpdateLastactivity

Update last activity for the connection after receiving keep-alive (KA) probes.

Possible values: ENABLED, DISABLED

Default value: ENABLED

flavor

Set TCP congestion control algorithm.

Possible values: Default, Westwood

Default value: NS_TCP_DEFAULT

dynamicReceiveBuffering

Enable or disable dynamic receive buffering. When enabled, allows the receive buffer to be adjusted dynamically based on memory and network conditions.

Note: The buffer size argument must be set for dynamic adjustments to take place.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KA

Send periodic TCP keep-alive (KA) probes to check if peer is still up.

Possible values: ENABLED, DISABLED

Default value: DISABLED

KAconnIdleTime

Duration, in seconds, for the connection to be idle, before sending a keep-alive (KA) probe.

Default value: NSTCP_KA_DEFAULT_CONN_IDLETIME

Minimum value: 1

Maximum value: 4095

KAmaxProbes

Number of keep-alive (KA) probes to be sent when not acknowledged, before assuming the peer to be down.

Default value: NSTCP_KA_DEFAULT_PROBE_COUNT

Minimum value: 1

Maximum value: 255

KAprobeInterval

Time interval, in seconds, before the next keep-alive (KA) probe, if the peer does not respond.

Default value: NSTCP_KA_DEFAULT_INTERVAL

Minimum value: 1

Maximum value: 4095

sendBuffsize

TCP Send Buffer Size

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

mptcp

Enable/Disable Multi-Path TCP

Possible values: ENABLED, DISABLED

Default value: DISABLED

EstablishClientConn

Establishing Client Client connection on First data/ Final-ACK / Automatic

Possible values: AUTOMATIC, CONN_ESTABLISHED, ON_FIRST_DATA

Default value: NS_CONN_AUTOMATIC

Example

```
set tcpprofile <profile name> -WS ENABLED -WSVAL 4
```

---

unset ns tcpProfile

Removes the attributes of the TCP profile. Attributes for which a default value is available revert to their default values. Refer to the 'set ns tcpProfile' command for a description of the parameters..Refer to the set ns tcpProfile command for

meanings of the arguments.

Synopsys

unset ns tcpProfile <name> [-WS] [-SACK] [-WSVal] [-nagle] [-ackOnPush] [-mss] [-maxBurst] [-initialCwnd] [-delayedAck] [-oooQSize] [-maxPktPerMss] [-pktPerRetx] [-minRTO] [-slowStartIncr] [-bufferSize] [-synCookie] [-KAprobeUpdateLastactivity] [-flavor] [-dynamicReceiveBuffering] [-KA] [-KAmaxProbes] [-KAconnIdleTime] [-KAprobeInterval] [-sendBuffsize] [-mptcp] [-EstablishClientConn]

---

show ns tcpProfile

Displays information about TCP profiles configured on the appliance.

Synopsys

show ns tcpProfile [<name>]

Arguments

name

Name of the TCP profile to be displayed. If a name is not provided, information about all TCP profiles is shown.

summary

fullValues

format

level

Outputs

WS

Enable or disable window scaling.

SACK

Enable or disable Selective ACKnowledgement (SACK).

WSVal

Factor used to calculate the new window size.

This argument is needed only when window scaling is enabled.

nagle

Enable or disable the Nagle algorithm on TCP connections.

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing

Web 2.0 PUSH.

mss

Maximum Segment Size(MSS) to use for TCP Connection(0 forces use of global setting)

maxBurst

Maximum number of TCP segments allowed in a burst.

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means infinite.

maxPktPerMss

Maximum packet per MSS value

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

minRTO

TCP minimum RTO (in millisec)

slowStartIncr

TCP slowstart increment factor

bufferSize

TCP Buffer size

flavor

TCP algorithm

refCnt

Number of entities using this profile

synCookie

Enable or disable the SYNCOOKIE mechanism for TCP handshake with clients. Disabling SYNCOOKIE prevents SYN attack protection on the NetScaler appliance.

KAprobeUpdateLastactivity

Update last activity for the connection after receiving keep-alive (KA) probes.

dynamicReceiveBuffering

Enable or disable dynamic receive buffering. When enabled, allows the receive buffer to be adjusted dynamically based on memory and network conditions.

Note: The buffer size argument must be set for dynamic adjustments to take place.

KA

Send periodic TCP keep-alive (KA) probes to check if peer is still up.

KAconnIdleTime

Duration, in seconds, for the connection to be idle, before sending a keep-alive (KA) probe.

KAmaxProbes

Number of keep-alive (KA) probes to be sent when not acknowledged, before assuming the peer to be down.

KAprobeInterval

Time interval, in seconds, before the next keep-alive (KA) probe, if the peer does not respond.

sendBuffsize

TCP Send Buffer size

mptcp

Enable/Disable Multi-Path TCP

EstablishClientConn

Allocating Client Connection On

stateflag

State flag

devno

count

Example

```
show tcp profile [profile name]
```

ns tcpbufParam

Sep 22, 2015

The following operations can be performed on "ns tcpbufParam":

set | unset | show

set ns tcpbufParam

Sets the attributes for the TCP buffering per connection.

Synopsys

set ns tcpbufParam [-size <KBytes>] [-memLimit <MBytes>]

Arguments

size

TCP buffering size per connection, in kilobytes.

Default value: 64

Minimum value: 4

Maximum value: 20480

memLimit

Maximum memory, in megabytes, that can be used for buffering.

Default value: 64

---

unset ns tcpbufParam

Use this command to remove ns tcpbufParam settings.Refer to the set ns tcpbufParam command for meanings of the arguments.

Synopsys

unset ns tcpbufParam [-size] [-memLimit]

---

show ns tcpbufParam

Displays the TCP buffering configuration on the appliance.

Synopsys

show ns tcpbufParam

Arguments

format

level

Outputs

size

TCP buffering size per connection, in kilobytes.

memLimit

Maximum memory, in megabytes, that can be used for buffering.

Example

An example of this command's output is as follows: TCP buffer size: 64KBytes TCP buffer percentage: 50%

ns timeout

Sep 22, 2015

The following operations can be performed on "ns timeout":

set | unset | show

set ns timeout

Sets timeout values for various aspects of the NetScaler appliance. Caution: Modifying these values can affect system performance.

Synopsys

set ns timeout [-zombie <positive_integer>] [-httpClient <positive_integer>] [-httpServer <positive_integer>] [-tcpClient <positive_integer>] [-tcpServer <positive_integer>] [-anyClient <positive_integer>] [-anyServer <positive_integer>] [-halfclose <positive_integer>] [-nontcpZombie <positive_integer>] [-ReducedFinTimeOut <positive_integer>] [-NewConnIdleTimeOut <positive_integer>]

Arguments

zombie

Interval, in seconds, at which the NetScaler zombie cleanup process must run. This process cleans up inactive TCP connections.

Default value: 120

Minimum value: 1

Maximum value: 600

client

Client idle timeout (in seconds). If zero, the service-type default value is taken when service is created.

Maximum value: 18000

server

Server idle timeout (in seconds). If zero, the service-type default is taken when service is created.

Maximum value: 18000

httpClient

Global idle timeout, in seconds, for client connections of HTTP service type. This value is over ridden by the client timeout that is configured on individual entities.

Maximum value: 18000

httpServer

Global idle timeout, in seconds, for server connections of HTTP service type. This value is over ridden by the server timeout that is configured on individual entities.

Maximum value: 18000

tcpClient

Global idle timeout, in seconds, for non-HTTP client connections of TCP service type. This value is over ridden by the client timeout that is configured on individual entities.

Maximum value: 18000

tcpServer

Global idle timeout, in seconds, for non-HTTP server connections of TCP service type. This value is over ridden by the server timeout that is configured on entities.

Maximum value: 18000

anyClient

Global idle timeout, in seconds, for non-TCP client connections. This value is over ridden by the client timeout that is configured on individual entities.

Maximum value: 31536000

anyServer

Global idle timeout, in seconds, for non TCP server connections. This value is over ridden by the server timeout that is configured on individual entities.

Maximum value: 31536000

halfclose

Idle timeout, in seconds, for connections that are in TCP half-closed state.

Default value: 10

Minimum value: 1

Maximum value: 600

nontcpZombie

Interval at which the zombie clean-up process for non-TCP connections should run. Inactive IP NAT connections will be cleaned up.

Default value: 60

Minimum value: 1

Maximum value: 600

ReducedFinTimeOut

Alternative idle timeout for new TCP NATPCB connections.

Default value: 30

Minimum value: 1

Maximum value: 300

NewConnIdleTimeOut

Timer interval(in seconds) for new NATPCB for tcp connections.

Default value: 4

Minimum value: 1

Maximum value: 120

Example

```
set ns timeout -zombie 200
```

---

unset ns timeout

Use this command to remove ns timeout settings.Refer to the set ns timeout command for meanings of the arguments.

Synopsys

```
unset ns timeout [-zombie] [-httpClient] [-httpServer] [-tcpClient] [-tcpServer] [-anyClient] [-anyServer] [-halfclose] [-nontcpZombie] [-ReducedFinTimeOut] [-NewConnIdleTimeOut]
```

---

show ns timeout

Displays the timeouts configured for various NetScaler entities. Note: The timeouts having default values are not displayed.

Synopsys

```
show ns timeout
```

Arguments

format

level

Outputs

zombie

Timer interval(in seconds) for zombie process that cleanup inactive TCP connections

Minimum value: 1

Maximum value: 600

Default value: 120

client

Client idle timeout (in seconds). If zero, the service-type default value is taken when service is created.

server

Server idle timeout (in seconds). If zero, the service-type default is taken when service is created.

httpClient

HTTP client idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

httpServer

HTTP server idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

tcpClient

TCP client idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

tcpServer

TCP server idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

anyClient

ANY client idle timeout (in seconds)

Minimum value: 0

Maximum value: 31536000

anyServer

ANY server idle timeout (in seconds)

Minimum value: 0

Maximum value: 31536000

halfclose

Half-closed connection timeout (in seconds)

Minimum value: 1

Maximum value: 600

Default value: 10

nontcpZombie

Timer interval(in seconds) for zombie process that cleanup inactive IP NAT connections

Minimum value: 1

Maximum value: 600

Default value: 60

ReducedFinTimeOut

Timer interval(in seconds) for NATPCB for tcp flow

NewConnIdleTimeOut

Timer interval(in seconds) for new NATPCB for tcp connections

Example

```
show ns timeout
```

ns timer

Sep 22, 2015

The following operations can be performed on "ns timer":

add | rm | set | unset | bind | unbind | show | rename

add ns timer

Create a Timer.

Synopsys

add ns timer <name> (-interval <integer> [<unit>]) [-comment <string>]

Arguments

name

Timer name.

interval

The frequency at which the policies bound to this timer are invoked. The minimum value is 20 msec. The maximum value is 20940 in seconds and 349 in minutes

Default value: 5

Minimum value: 1

Maximum value: 20940000

comment

Comments associated with this timer.

Example

```
add timer policy timer -comment "Timer that would be invoked at interval 10 sec apart."
```

---

rm ns timer

Remove a Timer.

Synopsys

rm ns timer <name>

Arguments

name

Timer name.

Example

```
rm ns timer timer
```

---

set ns timer

Set a argument values for existing timer.

Synopsys

set ns timer <name> [-interval <integer>] [<unit>] [-comment <string>]

Arguments

name

Timer name.

interval

The frequency at which the policies bound to this timer are invoked. The minimum value is 20 msec. The maximum value is 20940 in seconds and 349 in minutes

Default value: 5

Minimum value: 1

Maximum value: 20940000

unit

Timer interval unit

Possible values: SEC, MIN

Default value: NSTMUNT_SEC

comment

Comments associated with this timer.

Example

```
set ns timer timer -comment "Timer that would be invoked at interval 20 sec apart."
```

---

unset ns timer

Unset comment for existing timer..Refer to the set ns timer command for meanings of the arguments.

Synopsys

unset ns timer <name> [-interval <integer>] [<unit>] [-comment <string>]

Example

```
unset ns timer timer -comment
```

---

bind ns timer

Defines the binding relation among timer, and timer policy.

Synopsys

bind ns timer <name> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-vServer <string>] [-sampleSize <positive_integer>] [-threshold <positive_integer>]

Arguments

name

Timer name.

policyName

The timer policy associated with the timer.

Example

i) bind ns timer timer_trigger -policyName timer_pol -priority 1 ii) bind ns timer timer_trigger -policyName timer_pol -priority 1

---

unbind ns timer

Unbind entities from timer

Synopsys

unbind ns timer <name> -policyName <string>

Arguments

name

Timer name.

policyName

The timer policy associated with the timer.

Example

unbind ns timer timer -policyName timer_pol

---

show ns timer

Display the Timer entities.

Synopsys

show ns timer [<name>]

Arguments

name

Timer name.

summary

fullValues

format

level

Outputs

interval

The frequency at which the policies bound to this timer are invoked. The minimum value is 20 msec. The maximum value is 20940 in seconds and 349 in minutes

unit

Timer interval unit

comment

Comments associated with this timer.

policyName

The timer policy associated with the timer.

priority

Specifies the priority of the timer policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

vServer

Name of the vserver which provides the context for the rule in timer policy. When not specified it is treated as a Global Default context.

sampleSize

Denotes the sample size. Sample size value of 'x' means that previous '(x - 1)' policy's rule evaluation results and the current evaluation result are present with the binding. For example, sample size of 10 means that there is a state of previous 9 policy evaluation results and also the current policy evaluation result.

threshold

Denotes the threshold. If the rule of the policy in the binding relation evaluates 'threshold size' number of times in 'sample size' to true, then the corresponding action is taken. Its value needs to be less than or equal to the sample size value.

stateflag

bindPolicyType

devno

count

---

rename ns timer

Rename a timer.

Synopsys

rename ns timer <name>@ <newName>@

Arguments

name

The name of the timer.

newName

The new name of the timer.

Example

rename ns timer oldname newname

---

ns trafficDomain

Sep 22, 2015

The following operations can be performed on "ns trafficDomain":

add | rm | clear | bind | unbind | enable | disable | show | stat

add ns trafficDomain

Configure Traffic Domain on the system.

Synopsys

add ns trafficDomain <td> [-aliasName <string>]

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

aliasName

Name of traffic domain being added.

Example

```
add ns trafficDomain 1 -aliasName td1
```

---

rm ns trafficDomain

Remove Traffic Domain configured.

Synopsys

rm ns trafficDomain <td>

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

Example

```
rm ns trafficDomain 1
```

---

clear ns trafficDomain

Remove Traffic Domain configuration.

Synopsys

clear ns trafficDomain <td>

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

---

bind ns trafficDomain

bind vlan or bridgegroup entities with traffic domain.

Synopsys

bind ns trafficDomain <td> [-vlan <positive_integer>] [-bridgegroup <positive_integer>]

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

vlan

Names of all member vlans.

Minimum value: 1

Maximum value: 4094

bridgegroup

bridge group.

Minimum value: 1

Maximum value: 1000

Example

```
bind ns trafficDomain 1 -vlan 2
```

---

unbind ns trafficDomain

Unbind vlan or bridgegroup entities from traffic domain

Synopsys

```
unbind ns trafficDomain <td> [-vlan <positive_integer>] [-bridgegroup <positive_integer>]
```

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

vlan

Names of all member vlans.

Minimum value: 1

Maximum value: 4094

bridgegroup

bridge group.

Minimum value: 1

Maximum value: 1000

Example

```
unbind ns trafficDomain 1 -vlan 2
```

---

enable ns trafficDomain

Enable TrafficDomain.

Synopsys

```
enable ns trafficDomain <td> [-state ( ENABLED | DISABLED )]
```

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

state

The state of TrafficDmain.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
enable ns trafficdomain 1
```

---

disable ns trafficDomain

Disable TrafficDomain.

Synopsys

```
disable ns trafficDomain <td> [-state ( ENABLED | DISABLED )]
```

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

state

The state of TrafficDmain.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
disable ns trafficdomain 1
```

---

show ns trafficDomain

Display Traffic Domain configuration.

Synopsys

```
show ns trafficDomain [<td>]
```

Arguments

td

Traffic Domain Id.

Minimum value: 1

Maximum value: 4094

format

level

Outputs

aliasName

Name of traffic domain being added.

stateflag

Used internally for display.

vlan

Names of all member vlans.

bridgegroup

bridge group.

state

The state of TrafficDmain.

devno

count

Example

An example of the output of the show trafficDomain command is as follows: 1) Td: 1 Alias Name: State: ENABLED Vlans : 50 2) Td: 2 Ali

---

stat ns trafficDomain

Display statistics for Traffic Domains(s).

Synopsys

stat ns trafficDomain [<td>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

td

An integer specifying the Traffic Domain ID. Possible values: 1 through 4094.

Minimum value: 1

Maximum value: 4094

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received (TdRxPkts)

Packets received on this TD.

Packets sent (TdTxPkts)

Packets transmitted from this TD.

Packets dropped (TdDropPkts)

Inbound packets dropped on this TD by reception.

Example

stat ns trafficdomain 1

---

ns version

Sep 22, 2015

The following operations can be performed on "ns version":

show ns version

Displays the version and build number of the appliance.

Synopsys

show ns version

Outputs

version

Version.

Mode

Kernel mode (KMPE/VMPE).

ns weblogparam

Sep 22, 2015

The following operations can be performed on "ns weblogparam":

set | unset | show

set ns weblogparam

Sets the Weblog parameters.

Synopsys

set ns weblogparam [-bufferSizeMB <positive_integer>] [-customReqHdrs <string> ...] [-customRspHdrs <string> ...]

Arguments

bufferSizeMB

Buffer size, in MB, allocated for log transaction data on the system. The maximum value is limited to the memory available on the system. Note: For the change in buffer size to take effect, you must disable and then re-enable the Web Server Logging feature on the appliance.

Default value: 16

Minimum value: 1

Maximum value: 4294967294LU

customReqHdrs

Name(s) of HTTP request headers whose values should be exported by the Web Logging feature. A maximum of two header names can be configured, with each header name having a maximum length of 31 characters.

customRspHdrs

Name(s) of HTTP response headers whose values should be exported by the Web Logging feature. A maximum of two header names can be configured, with each header name having a maximum length of 31 characters.

---

unset ns weblogparam

Use this command to remove ns weblogparam settings.Refer to the set ns weblogparam command for meanings of the arguments.

Synopsys

unset ns weblogparam [-bufferSizeMB] [-customReqHdrs] [-customRspHdrs]

---

show ns weblogparam

Displays the Weblog parameters.

Synopsys

show ns weblogparam

Arguments

format

level

Outputs

bufferSizeMB

Buffer size in MB.

customReqHdrs

Name(s) of HTTP request headers whose values should be exported by the Web Logging feature.

customRspHdrs

Name(s) of HTTP response headers whose values should be exported by the Web Logging feature.

ns xmlnamespace

Sep 22, 2015

The following operations can be performed on "ns xmlnamespace":

add | rm | set | unset | show

add ns xmlnamespace

Adds a mapping between an XML prefix and a namespace URI (Uniform Resource Identifier).

Synopsys

add ns xmlnamespace <prefix> <namespace>

Arguments

prefix

XML prefix.

namespace

Expanded namespace for which the XML prefix is provided.

Example

```
add ns xmlnamespace soap http://schemas.xmlsoap.org/soap/envelope/
```

---

rm ns xmlnamespace

Removes the mapping between an XML prefix and a namespace URI.

Synopsys

rm ns xmlnamespace <prefix>

Arguments

prefix

XML prefix for which the mapping must be removed.

Example

```
rm ns xmlnamespace soap
```

---

set ns xmlnamespace

Modifies the mapping between an XML prefix and a namespace URI.

Synopsys

set ns xmlnamespace <prefix> [<namespace>] [-description <string>]

Arguments

prefix

XML prefix for which the namespace or description must be added or updated.

namespace

Expanded namespace for which the XML prefix is provided.

description

Description for the prefix.

Example

```
set ns xmlnamespace soap -description SOAP/1.1
```

---

unset ns xmlnamespace

Use this command to remove ns xmlnamespace settings.Refer to the set ns xmlnamespace command for meanings of the arguments.

Synopsys

unset ns xmlnamespace <prefix> [-namespace] [-description]

---

show ns xmlnamespace

Displays the mappings between XML prefixes to namespace URIs.

Synopsys

show ns xmlnamespace [<prefix>]

Arguments

prefix

Name of the prefix for which the mappings must be displayed.

summary

fullValues

format

level

Outputs

namespace

Expanded namespace for which the XML prefix is provided.

description

Description for the prefix.

devno

count

stateflag

Example

```
show ns xmlnamespace soap
```

reboot

Sep 22, 2015

The following operations can be performed on "reboot":

Restarts the NetScaler appliance. Note: * When a standalone NetScaler appliance is rebooted, the unsaved configurations (configurations performed since the last 'save ns config' command was issued) are lost. * In the high availability mode, when the primary appliance is rebooted, the secondary system takes over and becomes the primary. The unsaved configurations from the old primary are available on the new primary appliance. * In a cluster setup, this command can be executed only through the cluster IP address and it reboots only the configuration coordinator.

reboot [-warm]

warm

Restarts the NetScaler software without rebooting the underlying operating system. The session terminates and you must log on to the appliance after it has restarted.

Note: This argument is required only for nCore appliances. Classic appliances ignore this argument.

shutdown

Sep 22, 2015

The following operations can be performed on "shutdown":

Stops all operations and powers off the NetScaler appliance. Note: * When a standalone NetScaler appliance is shut down, the unsaved configurations (configurations performed since the last 'save ns config' command was issued) are lost. * In a high availability setup, when the primary appliance is shut down, the secondary appliance takes over and becomes the primary. The unsaved configurations from the old primary are available on the new primary appliance. * In a cluster setup, this command can be executed only through the cluster IP address and it shuts down only the configuration coordinator.

shutdown

NTP Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- ntp param
- ntp server
- ntp status
- ntp sync

ntp param

Sep 22, 2015

The following operations can be performed on "ntp param":

set | unset | show

Modifies the values for NTP parameters on the NetScaler appliance.

set ntp param [-authentication (YES | NO)] [-trustedkey <positive_integer> ...] [-autokeyLogsec <positive_integer>] [-revokeLogsec <positive_integer>]

authentication

Apply NTP authentication, which enables the NTP client (NetScaler) to verify that the server is in fact known and trusted.

Possible values: YES, NO

Default value: YES

trustedkey

Key identifiers that are trusted for server authentication with symmetric key cryptography in the keys file.

Minimum value: 1

Maximum value: 65534

autokeyLogsec

Autokey protocol requires the keys to be refreshed periodically. This parameter specifies the interval between regenerations of new session keys. In seconds, expressed as a power of 2.

Default value: 12

Maximum value: 32

revokeLogsec

Interval between re-randomizations of the autokey seeds to prevent brute-force attacks on the autokey algorithms.

Default value: 16

Maximum value: 32

Use this command to remove ntp param settings.Refer to the set ntp param command for meanings of the arguments.

unset ntp param [-authentication] [-trustedkey] [-autokeyLogsec] [-revokeLogsec]

Displays information about the NTP parameters.

show ntp param

format

level

authentication

Apply NTP authentication, which enables the NTP client (NetScaler) to verify that the server is in fact known and trusted.

trustedkey

Key identifiers that are trusted for server authentication with symmetric key cryptography in the keys file.

autokeyLogsec

Autokey protocol requires the keys to be refreshed periodically. This parameter specifies the interval between regenerations of new session keys. In seconds, expressed as a power of 2.

revokeLogsec

Interval between re-randomizations of the autokey seeds to prevent brute-force attacks on the autokey algorithms.

---

ntp server

Sep 22, 2015

The following operations can be performed on "ntp server":

add | rm | set | unset | show

Adds an NTP server to the appliance. This server can be used to synchronize the time on the appliance to the network time.

add ntp server (<serverIP> | <serverName>) [-minpoll <positive_integer>] [-maxpoll <positive_integer>] [-autokey | -key <positive_integer>]

serverIP

IP address of the NTP server.

serverName

Fully qualified domain name of the NTP server.

minpoll

Minimum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MINPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

maxpoll

Maximum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MAXPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

autokey

Use the Autokey protocol for key management for this server, with the cryptographic values (for example, symmetric key, host and public certificate files, and sign key) generated by the ntp-keygen utility. To require authentication for communication with the server, you must set either the value of this parameter or the key parameter.

key

Key to use for encrypting authentication fields. All packets sent to and received from the server must include authentication fields encrypted by using this key. To require authentication for communication with the server, you must set either the value of this parameter or the autokey parameter.

Minimum value: 1

Maximum value: 65534

Removes an NTP server. You can specify the server by IP address or by name.

rm ntp server (<serverIP> | <serverName>)

serverIP

IP address of the NTP server to be removed.

serverName

Name of the NTP server to be removed.

Modifies the specified attributes of an NTP server.

set ntp server (<serverIP> | <serverName>) [-minpoll <positive_integer>] [-maxpoll <positive_integer>] [-preferredNtpServer (YES | NO)] [-autokey | -key <positive_integer>]

serverIP

IP address of the NTP server to be modified.

serverName

Name of the NTP server to be modified.

minpoll

Minimum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MINPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

maxpoll

Maximum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MAXPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

preferredNtpServer

Preferred NTP server. The NetScaler appliance chooses this NTP server for time synchronization among a set of correctly operating hosts.

Possible values: YES, NO

Default value: NO

autokey

Use the Autokey protocol for key management for this server, with the cryptographic values (for example, symmetric key, host and public certificate files, and sign key) generated by the ntp-keygen utility. To require authentication for communication with the server, you must set either the value of this parameter or the key parameter.

key

Key to use for encrypting authentication fields. All packets sent to and received from the server must include authentication fields encrypted by using this key. To require authentication for communication with the server, you must set either the value of this parameter or the autokey parameter.

Minimum value: 1

Maximum value: 65534

Use this command to remove ntp server settings.Refer to the set ntp server command for meanings of the arguments.

unset ntp server [<serverIP>] [<serverName>] [-minpoll] [-maxpoll] [-preferredNtpServer] [-autokey] [-key]

Displays information about an NTP server. You can specify the server by IP address or by name.

show ntp server [<serverIP> | <serverName>]

serverIP

IP address of the NTP server about which to display information.

serverName

Name of the NTP server about which to display information.

summary

fullValues

format

level

minpoll

Minimum poll interval of the server in secs.

maxpoll

Maximum poll interval of the server in secs.

preferredNtpServer

Preferred NTP server. The NetScaler appliance chooses this NTP server for time synchronization among a set of correctly operating hosts.

autokey

Use the Autokey protocol for key management for this server, with the cryptographic values (for example, symmetric key, host and public certificate files, and sign key) generated by the ntp-keygen utility. To require authentication for communication with the server, you must set either the value of this parameter or the key parameter.

key

Key to use for encrypting authentication fields. All packets sent to and received from the server must include authentication fields encrypted by using this key. To require authentication for communication with the server, you must set either the value of this parameter or the autokey parameter.

devno

count

stateflag

ntp status

Sep 22, 2015

The following operations can be performed on "ntp status":

Displays the NTP status on the appliance.

show ntp status

response

ntp sync

Sep 22, 2015

The following operations can be performed on "ntp sync":

enable | disable | show

Enables NTP synchronization. When NTP synchronization is enabled, the NTP daemon is spawned for time synchronization.

enable ntp sync

Disables NTP synchronization.

disable ntp sync

Displays the status of the NTP synchronization.

show ntp sync

state

Show NTP status

Policy Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- policy dataset
- policy evaluation
- policy expression
- policy httpCallout
- policy map
- policy patClass
- policy patset
- policy stringmap

policy dataset

Sep 22, 2015

The following operations can be performed on "policy dataset":

add | rm | bind | unbind | show

Adds a policy dataset to the appliance.

add policy dataset <name> <type> [-indexType (Auto-generated | User-defined)]

name

Name of the dataset. Must not exceed 127 characters.

type

Type of value to bind to the dataset.

Possible values: ipv4, number, ipv6

indexType

Index type.

```
add policy dataset ts1 -type IPV4
```

Removes a dataset from the appliance.

rm policy dataset <name>

name

Name of the dataset to remove.

```
rm policy dataset pat1
```

Binds a value of the specified type to the dataset. If the first value is bound by using an index label, the other bind statements to that set should also provide an index.

bind policy dataset <name> <value> [-index <positive_integer>]

name

Name of the dataset to which to bind the value.

value

Value of the specified type that is associated with the dataset.

```
bind policy dataset ts1 192.168.20.1 -index 2
```

Unbind string(s) from a dataset.

unbind policy dataset <name> <value>

name

Name of the dataset from which to unbind the value.

value

Value to unbind from the dataset.

```
unbind policy dataset pat1 bar xyz
```

Display the configured dataset(s).

show policy dataset [<name>]

name

Name of the dataset. Must not exceed 127 characters.

summary

fullValues

format

level

stateflag

value

Value of the specified type that is associated with the dataset.

index

The index of the value (ipv4, ipv6, number) associated with the set.

description

Description of the set

type

Type of value to bind to the dataset.

indexType

Index type.

MaxIndex

Maximum number of values bounded to dataset. The maxindex value will not be decreased when we unbind a value from the dataset. This field is used in auto-generated indexing type.

devno

count

show policy dataset set1

policy evaluation

Sep 22, 2015

The following operations can be performed on "policy evaluation":

Executes pixl expression or action and gives result. Result type can be zero or more of: -Bool -Num -Double -Unsigned long -String

show policy evaluation (-expression <expression> | -action <string>) -type <type> -input <string>

expression

Expression string. For example: http.req.body(100).contains("this").

action

Rewrite action name. Supported rewrite action types are:

-delete

-delete_all

-delete_http_header

-insert_after

-insert_after_all

-insert_before

-insert_before_all

-insert_http_header

-replace

-replace_all

type

Indicates request or response input packet

Possible values: HTTP_REQ, HTTP_RES, TEXT

input

Text representation of input packet.

summary

fullValues

stateflag

pitModifiedInputData

Text representation of packet after evaluating expression or rewrite action.

pitBoolResult

Result of the expression in bool format.

pitNumResult

Result of the expression in num format.

pitDoubleResult

Result of the expression in double format.

pitUlongResult

Result of the expression in unsigned long format.

pitRefResult

Result of the expression in string format.

isPitEmptyRefResult

Result of the expression is empty string.

pitOffsetResult

Offset of the resultant sting.

pitOffsetResultLen

Offset length of the resultant sting.

isTruncatedRefResult

Identify whether ref result is truncated result.

pitBoolEvalTime

Average evaluation time of bool type expression in nanoseconds.

pitNumEvalTime

Average evaluation time of num type expression in nanoseconds.

pitDoubleEvalTime

Average evaluation time of double type expression in nanoseconds.

pitUlongEvalTime

Average evaluation time of unsigned long type expression in nanoseconds.

pitRefEvalTime

Average evaluation time of string type expression in nanoseconds.

pitOffsetEvalTime

Average evaluation time in finding offset of the resultant string in the input. Time is in nanoseconds.

pitActionEvalTime

Average evaluation time of rewrite action in nanoseconds.

pitOperationPerformerArray

Details of the operation NS performed at various offsets during applying of rewrite action on input data. Operation can be insertion, modfication or deletion.

pitOldOffsetArray

Details of the offsets in the input data at which NS either inserted or modified or deleted data during applying of rewrite action.

pitNewOffsetArray

Details of the offsets in the output data at which NS either inserted or modified or deleted data during applying of rewrite action.

pitOffsetLengthArray

Details of the lengths of the data which NS either inserted or modified or deleted during applying of rewrite action.

pitBoolErrorResult

Result of the bool type expression if any error occurs during evaluation. Result will be in string format.

pitNumErrorResult

Result of the num type expression if any error occurs during evaluation. Result will be in string format.

pitDoubleErrorResult

Result of the double type expression if any error occurs during evaluation. Result will be in string format.

pitUlongErrorResult

Result of the unsigned long type expression if any error occurs during evaluation. Result will be in string format.

pitRefErrorResult

Result of the ref type expression if any error occurs during evaluation. Result will be in string format.

pitOffsetErrorResult

Result of the expression if any error occurs in calculating offset. Result will be in string format.

pitActionErrorResult

Result of the action if any error occurs in evaluation. Result will be in string format.

devno

count

Example 1: show policy evaluation -action rw_act_1 -type http_req -input 'GET / HTTP/1.1\\\\r\\\\nHost: abc.com\\\\r\\\\n\\\\r\\\\n' Example 2: show policy

policy expression

Sep 22, 2015

The following operations can be performed on "policy expression":

add | rm | set | unset | show

Creates a classic or default syntax named expression, which can be used in multiple policies. For example, you can create the following named expressions, ExpressionA and ExpressionB: ExpressionA: http.req.body(100).contains("A") ExpressionB: http.req.body(100).contains("B") You could then create an expression of the form: <ExpressionA || ExpressionB>

add policy expression <name> <value> [-comment <string>] [-clientSecurityMessage <string>]

name

Unique name for the expression. Not case sensitive. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Must not begin with 're' or 'xp' or be a word reserved for use as a default syntax expression qualifier prefix (such as HTTP) or enumeration value (such as ASCII). Must not be the name of an existing named expression, pattern set, dataset, stringmap, or HTTP callout.

value

Expression string. For example: http.req.body(100).contains("this").

description

Description for the expression.

comment

Any comments associated with the expression. Displayed upon viewing the policy expression.

clientSecurityMessage

Message to display if the expression fails. Allowed for classic end-point check expressions only.

Removes a named policy expression. If the expression is used by a policy or filter, you must remove the policy or filter before removing the expression.

rm policy expression <name> ...

name

Name of the policy expression to be removed.

Modifies the attributes of a named policy expression.

```
set policy expression <name> [<value>] [-comment <string>] [-clientSecurityMessage <string>]
```

name

Name of the policy expression to be modified.

value

The expression string.

description

Description for the expression.

comment

Any comments associated with the expression. Displayed upon viewing the policy expression.

clientSecurityMessage

The client security message that will be displayed on failure of this expression. Only relevant for end point check expressions.

Use this command to remove policy expression settings.Refer to the set policy expression command for meanings of the arguments.

```
unset policy expression <name> [-comment] [-clientSecurityMessage]
```

Displays information about the available named policy expressions.

show policy expression [<name> | -type (CLASSIC | ADVANCED)]

name

Name of the policy expression to display. If a name is not provided, information about all policy expressions is shown.

type

Type of expression. Can be a classic or default syntax (advanced) expression.

Possible values: CLASSIC, ADVANCED

summary

fullValues

format

level

value

The expression string.

hits

The total number of hits.

piHits

The total number of hits.

type

The type of expression. This is for output only.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

description

Description for the expression.NOTE: This attribute is deprecated.

comment

Any comments associated with the expression. Displayed upon viewing the policy expression.

stateflag

flag

isDefault

A value of true is returned if it is a default policy expression.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

policy httpCallout

Sep 22, 2015

The following operations can be performed on "policy httpCallout":

add | rm | set | unset | show

Adds a default syntax expression element that, when evaluated, sends an HTTP request to a specified service and receives an HTTP response from the service. Can be used to obtain additional information for use in evaluating policy rules and other expressions. This command only creates the HTTP callout. Use the 'set policy httpCallout' command to configure it.

add policy httpCallout <name>

name

Name for the HTTP callout. Not case sensitive. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Must not begin with 're' or 'xp' or be a word reserved for use as a default syntax expression qualifier prefix (such as HTTP) or enumeration value (such as ASCII). Must not be the name of an existing named expression, pattern set, dataset, stringmap, or HTTP callout.

```
add policy httpcallout h1
```

Removes an HTTP callout. You cannot remove an HTTP callout that is used in any part of policy, action, or expression.

rm policy httpCallout <name>

name

Name of the HTTP callout to remove.

```
rm policy httpcallout h1
```

Configures an HTTP callout element. You can then invoke the HTTP callout from other policies and functions (for example, actions) that use default syntax expressions. The expression prefix SYS.HTTP_CALLOUT invokes an HTTP callout. You can construct the HTTP callout request in one of two ways: * Specify individual parts of the request by using the HTTP method, host expression, URL stem expression, and header parameters. These parts are evaluated at run time and concatenated to build the request. * Specify the entire HTTP request in a single expression.

set policy httpCallout <name> [-IPAddress <ip_addr|ipv6_addr>] [-port <port>] [-vServer <string>] [-returnType <returnType>] [-httpMethod (GET | POST)] [-hostExpr <string>] [-urlStemExpr <string>] [-headers <name(value)> ...] [-parameters <name(value)> ...] [-bodyExpr <string>] [-fullReqExpr <string>] [-scheme (http | https)] [-resultExpr <string>] [-cacheForSecs <secs>]

name

Name of the HTTP callout to configure.

IPAddress

IP Address of the server (callout agent) to which the callout is sent. Can be an IPv4 or IPv6 address.

Mutually exclusive with the Virtual Server parameter. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

port

Server port to which the HTTP callout agent is mapped. Mutually exclusive with the Virtual Server parameter. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

vServer

Name of the load balancing, content switching, or cache redirection virtual server (the callout agent) to which the HTTP callout is sent. The service type of the virtual server must be HTTP. Mutually exclusive with the IP address and port parameters. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

returnType

Type of data that the target callout agent returns in response to the callout.

Available settings function as follows:

* TEXT - Treat the returned value as a text string.

* NUM - Treat the returned value as a number.

* BOOL - Treat the returned value as a Boolean value.

Note: You cannot change the return type after it is set.

Possible values: BOOL, NUM, TEXT

httpMethod

Method used in the HTTP request that this callout sends. Mutually exclusive with the full HTTP request expression.

Possible values: GET, POST

hostExpr

Default Syntax string expression to configure the Host header. Can contain a literal value (for example, 10.101.10.11) or a derived value (for example, http.req.header("Host")). The literal value can be an IP address or a fully qualified domain name. Mutually exclusive with the full HTTP request expression.

urlStemExpr

Default Syntax string expression for generating the URL stem. Can contain a literal string (for example, "/mysite/index.html") or an expression that derives the value (for example, http.req.url). Mutually exclusive with the full HTTP request expression.

headers

One or more headers to insert into the HTTP request. Each header is specified as "name(expr)", where expr is a default syntax expression that is evaluated at runtime to provide the value for the named header. You can configure a maximum of eight headers for an HTTP callout. Mutually exclusive with the full HTTP request expression.

parameters

One or more query parameters to insert into the HTTP request URL (for a GET request) or into the request body (for a POST request). Each parameter is specified as "name(expr)", where expr is an default syntax expression that is evaluated at run time to provide the value for the named parameter (name=value). The parameter values are URL encoded. Mutually exclusive with the full HTTP request expression.

bodyExpr

An advanced string expression for generating the body of the request. The expression can contain a literal string or an expression that derives the value (for example, client.ip.src). Mutually exclusive with -fullReqExpr.

fullReqExpr

Exact HTTP request, in the form of a default syntax expression, which the NetScaler appliance sends to the callout agent. If you set this parameter, you must not include HTTP method, host expression, URL stem expression, headers, or parameters.

The request expression is constrained by the feature for which the callout is used. For example, an HTTP.RES expression cannot be used in a request-time policy bank or in a TCP content switching policy bank.

The NetScaler appliance does not check the validity of this request. You must manually validate the request.

scheme

Type of scheme for the callout server.

Possible values: http, https

resultExpr

Expression that extracts the callout results from the response sent by the HTTP callout agent. Must be a response based expression, that is, it must begin with HTTP.RES. The operations in this expression must match the return type. For example, if you configure a return type of TEXT, the result expression must be a text based expression. If the return type is NUM, the result expression (resultExpr) must return a numeric value, as in the following example: http.res.body(10000).length.

cacheForSecs

Duration, in seconds, for which the callout response is cached. The cached responses are stored in an integrated caching content group named "calloutContentGroup". If no duration is configured, the callout responses will not be cached unless normal caching configuration is used to cache them. This parameter takes precedence over any normal caching configuration that would otherwise apply to these responses.

Note that the calloutContentGroup definition may not be modified or removed nor may it be used with other cache policies.

Minimum value: 1

Maximum value: 31536000

set policy httpcallout h1 -vServer v1

Use this command to remove policy httpCallout settings.Refer to the set policy httpCallout command for meanings of the arguments.

unset policy httpCallout <name> [-IPAddress] [-port] [-vServer] [-httpMethod] [-hostExpr] [-urlStemExpr] [-headers] [-parameters] [-bodyExpr] [-fullReqExpr] [-resultExpr] [-cacheForSecs]

Displays information about the configured HTTP callouts.

show policy httpCallout [<name>]

name

Name of the HTTP callout to display. If a name is not provided, information about all configured HTTP callouts is shown.

summary

fullValues

format

level

stateflag

IPAddress

Server IP address.

port

Server port.

vServer

Vserver name

returnType

Return type of the http callout

scheme

Type of scheme(HTTP/HTTPS) for the callout server

httpMethod

Http callout request type

hostExpr

PI string expression for Host

urlStemExpr

PI string expression for URL stem

headers

PI string expression for request http headers

parameters

PI string expression for request query parameters

fullReqExpr

PI string expression for full http callout request

resultExpr

PI string expression for http callout response

hits

Total hits

undefHits

Total undefs

svrState

The state of the service

undefReason

Reason for last undef

recursiveCallout

Number of recursive callouts

bodyExpr

An advanced string expression for generating the body of the request. The expression can contain a literal string or an expression that derives the value (for example, client.ip.src). Mutually exclusive with -fullReqExpr.

cacheForSecs

Duration, in seconds, for which the callout response is cached. The cached responses are stored in an integrated caching content group named "calloutContentGroup". If no duration is configured, the callout responses will not be cached unless normal caching configuration is used to cache them. This parameter takes precedence over any normal caching configuration that would otherwise apply to these responses.

Note that the calloutContentGroup definition may not be modified or removed nor may it be used with other cache policies.

devno

count

```
show policy httpcallout h1
```

policy map

Sep 22, 2015

The following operations can be performed on "policy map":

add | rm | show

Creates a policy to map a publicly known domain name to a target domain name for a reverse proxy virtual server used by the cache redirection feature. Optionally, you can also specify a source and target URL. The map policy can be associated with a reverse proxy cache redirection virtual server by using the 'bind cr vserver' command. There can be only one default map policy for a domain.

add policy map <mapPolicyName> -sd <string> [-su <string>] [-td <string>] [-tu <string>]

mapPolicyName

Name for the map policy. Must begin with a letter, number, or the underscore (_) character and must consist only of letters, numbers, and the hash (#), period (.), colon (:), space ( ), at (@), equals (=), hyphen (-), and underscore (_) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my map" or 'my map').

sd

Publicly known source domain name. This is the domain name with which a client request arrives at a reverse proxy virtual server for cache redirection. If you specify a source domain, you must specify a target domain.

su

Source URL. Specify all or part of the source URL, in the following format: /[[prefix] [*]] [.suffix].

td

Target domain name sent to the server. The source domain name is replaced with this domain name.

tu

Target URL. Specify the target URL in the following format: /[[prefix] [*]][.suffix].

Example 1 The following example creates a default map policy (map1) for the source domain www.a.com. Any client requests with this source domain in the host header is

---

Removes a map policy. Before removing the map policy, you must unbind the map policy from the reverse proxy virtual server.

rm policy map <mapPolicyName>

mapPolicyName

Name of the policy map to remove.

Displays information about the available policy maps.

show policy map [<mapPolicyName>]

mapPolicyName

Name of the policy map to display. If a name is not provided, information of all configured policy maps is shown.

summary

fullValues

format

level

sd

Publicly known source domain name. This is the domain name with which a client request arrives at a reverse proxy virtual server for cache redirection. If you specify a source domain, you must specify a target domain.

su

The source URL.

td

The domain name sent to the server.

tu

The target URL.

targetName

The expression string.

devno

count

stateflag

policy patClass

Sep 22, 2015

The following operations can be performed on "policy patClass":

add | rm | bind | unbind | show

Adds a pattern class. Each pattern class is identified by a name. More patterns (strings) can be associated with it later. NOTE: This command is deprecated. This command is deprecated in favor of 'add policy patset'. NOTE: This command is deprecated.This command is deprecated in favor of 'add policy patset'

name

Name of the pattern class. The name must not exceed 127 characters.

string

String associated with the patclass.

```
add policy patclass pat1 foo
```

Removes a pattern class. Once the pattern class is removed, all the expressions referring it have undefined values. NOTE: This command is deprecated in favor of 'rm policy patset'. NOTE: This command is deprecated.This command is deprecated in favor of 'rm policy patset'

name

Name of the pattern class.

```
rm policy patclass pat1
```

Binds string(s) to a pattern class. NOTE: This command is deprecated in favor of 'bind policy patset'. NOTE: This command is deprecated.This command is deprecated in favor of 'bind policy patset'

name

Name of the pattern class.

string

String associated with the patclass.

```
bind policy patclass pat1 bar xyz
```

Unbinds string(s) from a pattern class. NOTE: This command is deprecated in favor of 'unbind policy patset'. NOTE: This command is deprecated.This command is deprecated in favor of 'unbind policy patset'

name

Name of the pattern class.

string

String associated with the patclass.

```
unbind policy patclass pat1 bar xyz
```

Displays the configured pattern class(es). NOTE: This command is deprecated in favor of 'show policy patset'. NOTE: This command is deprecated.This command is deprecated in favor of 'show policy patset'

name

Name of the pattern class.

summary

fullValues

format

level

stateflag

string

String associated with the patclass.

index

The index of the string associated with the patclass.

charset

The character set of the string associated with patset.

description

Description of the patclass

isDefault

devno

count

show policy patclass pat1

policy patset

Sep 22, 2015

The following operations can be performed on "policy patset":

add | rm | bind | unbind | show

Adds a pattern set. A pattern set contains a name and one or more string patterns. Pattern sets can be used in default syntax expressions to match a set of strings. For example, HTTP.REQ.URL.EQUALS_ANY("test_urls"), where test_urls is a pattern set containing URL strings. Pattern sets can also be used in the search parameter of a rewrite action. Each string pattern is assigned an index that enables you to select the associated string from the set.

add policy patset <name> [-indexType (Auto-generated | User-defined)]

name

Unique name of the pattern set. Not case sensitive. Must begin with an ASCII letter or underscore (_) character and must contain only alphanumeric and underscore characters. Must not be the name of an existing named expression, pattern set, dataset, string map, or HTTP callout.

indexType

Index type.

add policy patset pat1

Removes a pattern set. If the pattern set is used by an expression in another object, such as a policy, you must remove the object before removing the pattern set.

rm policy patset <name>

name

Name of the pattern set to remove.

rm policy patset pat1

Binds a string to a pattern set.

bind policy patset <name> <string> [-index <positive_integer>] [-charset (ASCII | UTF_8)]

name

Name of the pattern set to which to bind the string.

string

String of characters that constitutes a pattern. For more information about the characters that can be used, refer to the character set parameter.

Note: Minimum length for pattern sets used in rewrite actions of type REPLACE_ALL, DELETE_ALL, INSERT_AFTER_ALL, and INSERT_BEFORE_ALL, is three characters.

bind policy patset pat1 bar -index 2

Unbinds a string from a pattern set.

unbind policy patset <name> <string> ...

name

Name of the pattern set from which to unbind a string.

string

String of characters to unbind from the pattern set.

unbind policy patset pat1 bar xyz

Displays the list of pattern sets configured on the appliance.

show policy patset [<name>]

name

Name of the pattern set for which to display the detailed information. If a name is not provided, a list of all pattern sets configured on the appliance is shown.

summary

fullValues

format

level

stateflag

string

String of characters that constitutes a pattern. For more information about the characters that can be used, refer to the character set parameter.

Note: Minimum length for pattern sets used in rewrite actions of type REPLACE_ALL, DELETE_ALL, INSERT_AFTER_ALL, and INSERT_BEFORE_ALL, is three characters.

index

The index of the string associated with the patset.

charset

Character set associated with the characters in the string.

Note: UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\\xNN'. For example, the UTF-8 character '?' can be encoded as '\\xC3\\xBC'.

description

Description of the patset

isDefault

indexType

Index type.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

MaxIndex

Maximum number of patterns bounded to pattern set. The maxindex value will not be decreased when we unbind a pattern from the patset. This field is used in auto-generated indexing type.

devno

count

show policy patset pat1

policy stringmap

Sep 22, 2015

The following operations can be performed on "policy stringmap":

add | rm | set | unset | bind | unbind | show

Creates a string map. You must use the 'bind policy stringmap' command to bind strings to this string map.

add policy stringmap <name> [-comment <string>]

name

Unique name for the string map. Not case sensitive. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Must not begin with 're' or 'xp' or be a word reserved for use as a default syntax expression qualifier prefix (such as HTTP) or enumeration value (such as ASCII). Must not be the name of an existing named expression, pattern set, dataset, string map, or HTTP callout.

comment

Comments associated with the string map.

i) add stringmap custom_stringmap . This creates a new string map with name custom_stringmap.

Removes a string map. String maps can be removed only if not used in any part of policy, action, or expression.

rm policy stringmap <name>

name

Name of the string map to remove.

i) rm stringmap custom_stringmap . This removes a string map whose name is custom_stringmap

---

Modifies the attributes of an existing string map.

set policy stringmap <name> -comment <string>

name

Name of the string map to be modified.

comment

Comments associated with the string map.

i) set stringmap custom_stringmap -comment "custom string map is for URLs." . This updates the comment associated with the string map whose name is custom_stringma

---

Use this command to remove policy stringmap settings.Refer to the set policy stringmap command for meanings of the arguments.

unset policy stringmap <name> -comment

Binds a key and its associated value to a string map. If the key already exists and has a different value, the old value is overwritten with the new value.

bind policy stringmap <name> <key> <value>

name

Name of the string map to which to bind the key-value pair.

key

Character string constituting the key to be bound to the string map. The key is matched against the data processed by the operation that uses the string map. The default character set is ASCII. UTF-8 characters can be included if the character set is UTF-8. UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\\xNN'. For example, the UTF-8 character '?' can be encoded as '\\xC3\\xBC'.

bind stringmap custom_stringmap "key-string" "value-string" . This adds the key "key-string" and its associated value "value-string" to the string map whose name is cus

---

Removes a key from the string map.

unbind policy stringmap <name> <key>

name

Name of the string map from which to remove a key.

key

Key to remove from the string map.

unbind stringmap custom_stringmap key1 . This removes the key "key1" and its associated value from the string map whose name is custom_stringmap.

Displays a list of available string maps.

show policy stringmap [<name>]

name

Name of the string map to display. If a name is not provided, a list of all the configured string maps is shown.

summary

fullValues

format

level

stateflag

comment

Comments associated with the string map.

key

Character string constituting the key to be bound to the string map. The key is matched against the data processed by the operation that uses the string map. The default character set is ASCII. UTF-8 characters can be included if the character set is UTF-8. UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\\xNN'. For example, the UTF-8 character '?' can be encoded as '\\xC3\\xBC'.

value

Character string constituting the value associated with the key. This value is returned when processed data matches the associated key. Refer to the key parameter for details of the value character set.

devno

count

show stringmap custom_stringmap . Displays all the key-value pairs of a string map whose name is custom-stringmap

PQ Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- pq
- pq binding
- pq policy
- pq stats

pq

Sep 22, 2015

The following operations can be performed on "pq":

Displays statistics of priority queuing.

stat pq [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Policy hits (PolMatch)

Number of times the Netscaler appliance matched an incoming request using any priority queuing policy.

Threshold failed (ThrsFail)

Number of times the Netscaler appliance failed to match an incoming request to any of priority queing policy.

Priority 1 requests (Pri1Req)

Number of priority 1 requests that the Netscaler appliance received.

Priority 2 requests (Pri2Req)

Number of priority 2 requests that the Netscaler appliance received.

Priority 3 requests (Pri3Req)

Number of priority 3 requests that the Netscaler appliance received.

pq binding

Sep 22, 2015

The following operations can be performed on "pq binding":

Displays the information about the priority queuing policy bound to the virtual server. NOTE: This command is deprecated.Deprecated as cluster dont support reverse binding

vServerName

Name of the load balancing virtual server for which to display the priority queuing policy.

summary

fullValues

stateflag

policyName

The name of the priority queuing policy.

rule

The condition for applying the policy.

priority

The priority of queuing the request.

weight

Weight.

qDepth

Queue Depth.

polqDepth

Policy Queue Depth.

hits

Total number of hits.

devno

count

pq policy

Sep 22, 2015

The following operations can be performed on "pq policy":

add | rm | set | unset | show | stat

Adds a priority queuing policy to the appliance. Note: To use the priority queuing policy on a virtual server, the virtual server must have priority queuing enabled and the priority queuing policy must be bound to the load balancing virtual server. To enable priority queuing on the virtual server and to bind the policy, use the set lb vserver and bind lb vserver commands.

add pq policy <policyName> -rule <expression> -priority <positive_integer> [-weight <positive_integer>] [-qDepth <positive_integer> | -polqDepth <positive_integer>]

policyName

Name for the priority queuing policy. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), and colon (:) characters.

rule

Expression or name of a named expression, against which the request is evaluated. The priority queuing policy is applied if the rule evaluates to true.

Note:

* On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you will not have to escape the double quotation marks.

* Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. Enter the value of positive_integer as 1, 2 or 3. The highest priority level is 1 and the lowest priority value is 3.

Minimum value: 1

Maximum value: 3

weight

Weight of the priority. Each priority is assigned a weight according to which it is served when server resources are available. The weight for a higher priority request must be set higher than that of a lower priority request.

To prevent delays for low-priority requests across multiple priority levels, you can configure weighted queuing for serving requests. The default weights for the priorities

are:

* Gold - Priority 1 - Weight 3

* Silver - Priority 2 - Weight 2

* Bronze - Priority 3 - Weight 1

Specify the weights as 0 through 101. A weight of 0 indicates that the particular priority level should be served only when there are no requests in any of the priority queues.

A weight of 101 specifies a weight of infinity. This means that this priority level is served irrespective of the number of clients waiting in other priority queues.

Maximum value: 101

qDepth

Queue depth threshold value. When the queue size (number of requests in the queue) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests in all the queues belonging to this policy) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

Removes a priority queuing policy from the appliance.

```
rm pq policy <policyName> ...
```

policyName

Name of the priority queuing policy to be removed.

Modifies the attributes of a priority queuing policy.

set pq policy <policyName> [-weight <positive_integer>] [-qDepth <positive_integer> | -polqDepth <positive_integer>]

policyName

Name of the priority queuing policy to be modified.

weight

Weight of the priority. Each priority is assigned a weight according to which it is served when server resources are available. The weight for a higher priority request must be set higher than that of a lower priority request.

To prevent delays for low-priority requests across multiple priority levels, you can configure weighted queuing for serving requests. The default weights for the priorities

are:

* Gold - Priority 1 - Weight 3

* Silver - Priority 2 - Weight 2

* Bronze - Priority 3 - Weight 1

Specify the weights as 0 through 101. A weight of 0 indicates that the particular priority level should be served only when there are no requests in any of the priority queues.

A weight of 101 specifies a weight of infinity. This means that this priority level is served irrespective of the number of clients waiting in other priority queues.

Maximum value: 101

qDepth

Queue depth threshold value. When the queue size (number of requests in the queue) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests in all the queues belonging to

this policy) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Maximum value: 4294967294

Use this command to remove pq policy settings.Refer to the set pq policy command for meanings of the arguments.

unset pq policy <policyName> [-weight] [-qDepth] [-polqDepth]

Displays information about the priority queuing policy.

show pq policy [<policyName>]

policyName

Name of the priority queuing policy about which to display information. If a name is not provided, information about all priority queuing policies is shown.

summary

fullValues

format

level

stateflag

rule

The condition for applying the policy.

priority

The priority of queuing the request.

weight

Weight.

qDepth

Queue Depth.

polqDepth

Policy Queue Depth.

hits

Total number of hits.

devno

count

Displays statistics of the priority queuing policy.

stat pq policy [<policyName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

policyName

Name of the priority queuing policy whose statistics must be displayed. If a name is not provided, statistics of all priority queuing policies are shown.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Toatal queue wait time (QWaitTim)

Amount of time spent by priority queuing clients waiting in the priority queue.

Avg queue wait time (AvWtTime)

Average wait time for clients for this priority queuing

policy.

Avg clt transaction time (AvgTime)

Average time taken by a priority queuing client to complete its transaction for this priority queuing policy.

Vserver IP (VsIP)

IP address of the virtual server to which this priority queuing policy is bound.

Vserver port (VsPort)

Port number of the virtual server to which this priority queuing policy is bound.

Current queue depth (Qdepth)

Number of clients waiting currently for this priority queuing policy.

Current server connections (ServCons)

Current number of server connections established for serving clients for this priority queuing policy.

Server TCP connections (TotServCon)

Total number of server connections established for serving clients for this priority queuing policy.

Client requests dropped (Dropped)

Total number of dropped transactions for this priority queuing policy.

Client HTTP transactions (CltTrns)

Total number of client transactions for this priority queuing policy.

Queue depth (TotQLen)

Total number of waiting clients for this priority queuing policy.

Avg clt transaction time (us) (AvgTime)

Average time taken by a priority queuing client to complete its transaction for this priority queuing policy.

pq stats

Sep 22, 2015

The following operations can be performed on "pq stats":

show pq stats is an alias for stat pq

```
show pq stats - alias for 'stat pq'
```

Protocol Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- protocol http
- protocol httpBand
- protocol icmp
- protocol icmpv6
- protocol ip
- protocol ipv6
- protocol tcp
- protocol udp

protocol http

Sep 22, 2015

The following operations can be performed on "protocol http":

Displays statistics of the HTTP protocol.

stat protocol http [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Total requests (HTReqRx)

Total number of HTTP requests received.

Total responses (HTRspRx)

Total number of HTTP responses sent.

Request bytes received (HTReqbRx)

Total number of bytes of HTTP request data received.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

GETs (HTGETs)

Total number of HTTP requests received with the GET method.

POSTs (HTPOSTs)

Total number of HTTP requests received with the POST method.

Other methods (HTOthers)

Total number of HTTP requests received with methods other than GET and POST. Some of the other well-defined HTTP methods are HEAD, PUT, DELETE, OPTIONS, and TRACE. User-defined methods are also allowed.

HTTP/1.0 requests (HT10ReqRx)

Total number of HTTP/1.0 requests received.

HTTP/1.1 requests (HT11ReqRx)

Total number of HTTP/1.1 requests received.

Content-length requests (HTCLnReq)

Total number of HTTP requests in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

Chunked requests (HTChkReq)

Total number of HTTP requests in which the Transfer-Encoding field of the HTTP header has been set to chunked.

Request bytes transmitted (HTReqbTx)

Total number of bytes of HTTP request data transmitted.

HTTP/1.0 responses (HT10RspRx)

Total number of HTTP/1.0 responses sent.

HTTP/1.1 responses (HT11RspRx)

Total number of HTTP/1.1 responses sent.

Content-length responses (HTCLnRsp)

Total number of HTTP responses sent in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

Chunked responses (HTChunk)

Total number of HTTP responses sent in which the Transfer-Encoding field of the HTTP header has been set to chunked. This setting is used when the server wants to start sending the response before knowing its total length. The server breaks the response into chunks and sends them in sequence, inserting the length of each chunk before the actual data. The message ends with a chunk of size zero.

Multi-part responses (HTMPrtHd)

Total number of HTTP multi-part responses sent. In multi-part responses, one or more entities are encapsulated within the body of a single message.

FIN-terminated responses (HTNoCLnChunk)

Total number of FIN-terminated responses sent. In FIN-terminated responses, the server finishes sending the data and closes the connection.

Response bytes transmitted (HTRspbTx)

Total number of bytes of HTTP response data transmitted.

Incomplete headers (HTIncHd)

Total number of HTTP requests and responses received in which the HTTP header spans more than one packet.

Incomplete request headers (HTIncReqHd)

Total number of HTTP requests received in which the header spans more than one packet.

Incomplete response headers (HTIncRspHd)

Total number of HTTP responses received in which the header spans more than one packet.

HTTP 500 Server-busy Responses (HT500Rsp)

Total number of HTTP error responses received. Some of the error responses are:

500 Internal Server Error

501 Not Implemented

502 Bad Gateway

503 Service Unavailable

504 Gateway Timeout

505 HTTP Version Not Supported

Large/Invalid messages (HTInvReq)

Total number of requests and responses received with large body.

Large/Invalid chunk requests (HTInvChkRx)

Total number of requests received with large chunk size, in which the Transfer-Encoding field of the HTTP header has been set to chunked.

Large/Invalid content-length (HTInvCLn)

Total number of requests received with large content, in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

SPDYv2 requests (SPDY2Strm)

Total number of requests received over SPDY

protocol httpBand

Sep 22, 2015

The following operations can be performed on "protocol httpBand":

set | unset | show

Sets the band size for HTTP request/response band statistics.

set protocol httpBand [-reqBandSize <integer>] [-respBandSize <integer>]

reqBandSize

Band size, in bytes, for HTTP request band statistics. For example, if you specify a band size of 100 bytes, statistics will be maintained and displayed for the following size ranges:

0 - 99 bytes

100 - 199 bytes

200 - 299 bytes and so on.

Default value: 100

Minimum value: 50

respBandSize

Band size, in bytes, for HTTP response band statistics. For example, if you specify a band size of 100 bytes, statistics will be maintained and displayed for the following size ranges:

0 - 99 bytes

100 - 199 bytes

200 - 299 bytes and so on.

Default value: 1024

Minimum value: 50

set protocol httpBand -reqBandSize 200 -respBandSize 2048

Use this command to remove protocol httpBand settings.Refer to the set protocol httpBand command for meanings of the arguments.

unset protocol httpBand [-reqBandSize] [-respBandSize]

Displays statistics of the HTTP request/response band.

show protocol httpBand -type (REQUEST | RESPONSE)

type

Type of statistics to display.

Possible values: REQUEST, RESPONSE

format

level

BandRange

The range of the HTTP request/response size, in bytes.

TotalBandSize

The total size of all HTTP requests/responses in this size range.

AvgBandSize

The average size of all HTTP requests/responses in this size range.

BandData

The total size of all HTTP requests/responses in this size range, expressed as a percentage of the total size of all HTTP requests/responses.

AccessCount

The number of HTTP requests/responses in this size range.

AccessRatio

The number of HTTP requests/responses in this size range, expressed as a percentage of the total number of HTTP requests/responses.

protocol icmp

Sep 22, 2015

The following operations can be performed on "protocol icmp":

Displays statistics of the ICMP protocol.

stat protocol icmp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

ICMP packets received (ICPktRx)

ICMP packets received.

ICMP bytes received (ICbRx)

Bytes of ICMP data received.

ICMP packets transmitted (ICPktTx)

ICMP packets transmitted.

ICMP bytes transmitted (ICbTx)

Bytes of ICMP data transmitted.

ICMP echo replies received (ECORepRx)

ICMP Ping echo replies received.

ICMP echo replies transmitted (ECORepTx)

ICMP Ping echo replies transmitted.

ICMP echos received (ECORx)

ICMP Ping Echo Request and Echo Reply packets received.

MTU lookup on dst ip info recvd (MTULkDst)

Total number of MTU lookup on destination IP info received on a need fragmentation ICMP error message failed.

ICMP rate threshold (pkts/sec) (ICThs)

Limit for ICMP packets handled every 10 milliseconds. Default value, 0, applies no limit.

This is a configurable value using the set rateControl command.

ICMP port unreachable received (PortUnRx)

ICMP Port Unreachable error messages received. This error is generated when there is no service is running on the port.

ICMP port unreachable generated (PortUnTx)

ICMP Port Unreachable error messages generated. This error is generated when there is no service is running on the port.

Need fragmentation received (NeedFrag)

ICMP Fragmentation Needed error messages received for packets that need to be fragmented but for which Don't Fragment is specified the header.

ICMP rate threshold exceeded (ICRtEx)

Times the ICMP rate threshold is exceeded. If this counter continuously increases, first make sure the ICMP packets received are genuine. If they are, increase the current rate threshold.

ICMP packets dropped (ICPktDr)

ICMP packets dropped because the rate threshold has been exceeded.

Bad ICMP checksum (BadCkSum)

ICMP Fragmentation Needed error messages received with an ICMP checksum error.

PMTU non-first IP fragments (PMTUerr)

ICMP Fragmentation Needed error messages received that were generated by an IP fragment other than the first one.

PMTU Invalid body len received (IvBdyLen)

ICMP Fragmentation Needed error messages received that specified an invalid body length.

PMTU no tcp connection (NoTcpCon)

ICMP Need Fragmentation error messages received for TCP packets. The state of the connection for these packets is not maintained on the NetScaler.

PMTU no udp conection (NoUdpCon)

ICMP Need Fragmentation error messages received for UDP packets. The state of the connection for these packets is not maintained on the NetScaler.

PMTU invalid tcp seqno recvd (InvSeqNo)

ICMP Fragmentation Needed error messages received for packets that contain an invalid TCP address.

Invalid next MTU value recvd (IvNxtMTU)

ICMP Fragmentation Needed error messages received in which the Maximum Transmission Unit (MTU) for the next hop is out of range. The range for the MTU is 576-1500.

Next MTU > Current MTU (BigNxMTU)

ICMP Fragmentation Needed error messages received in which the value for the next MTU is higher than that of the current MTU.

PMTU Invalid protocol recvd (IvPrtRx)

ICMP Fragmentation Needed error messages received that contain a protocol other than TCP and UDP.

PMTU IP check sum error (CkSumErr)

ICMP Fragmentation Needed error messages received with an IP checksum error.

PMTU pcb with no link (NoLnkErr)

ICMP Fragmentation Needed error messages received on a Protocol Control Block (PCB) with no link. The PCB maintains the state of the connection.

PMTU Discovery not enabled (PMTUdis)

ICMP Need Fragmentation error messages received when the PMTU Discovery mode is not enabled.

protocol icmpv6

Sep 22, 2015

The following operations can be performed on "protocol icmpv6":

Displays statistics of the ICMPv6 protocol.

stat protocol icmpv6 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

ICMPv6 packets received (icmpv6RxPkts)

ICMPv6 packets received.

ICMPv6 bytes received (icmpv6RxBytes)

Bytes of ICMPv6 data received.

ICMPv6 packets transmitted (icmpv6TxPkts)

ICMPv6 packets transmitted.

ICMPv6 bytes transmitted (icmpv6TxBytes)

Bytes of ICMPv6 data transmitted.

ICMPv6 NA packets received (icmpv6RxNa)

ICMPv6 neighbor advertisement packets received. These packets are received in response to a neighbor solicitation message sent out by this node, or if the link layer address of a neighbor has changed.

ICMPv6 NS packets received (icmpv6RxNs)

ICMPv6 neighbor solicitation packets received. These packets are received if the link layer address of a neighbor has changed, or in response to a neighbor solicitation message sent out by this node.

ICMPv6 RA packets received (icmpv6RxRa)

ICMPv6 router advertisement packets received. These are received at defined intervals or in response to a router solicitation message.

ICMPv6 RS packets received (icmpv6RxRs)

ICMPv6 router solicitation packets received. These could be sent by a neighboring router to initiate address resolution.

ICMPv6 Echo Request packets received (icmpv6RxEchoReq)

ICMPv6 Ping Echo Request packets received.

ICMPv6 Echo Reply packets received (icmpv6RxEchoReply)

ICMPv6 Ping Echo Reply packets received.

ICMPv6 NA packets transmitted (icmpv6TxNa)

ICMPv6 neighbor advertisement packets transmitted. These packets are sent in response to a neighbor solicitation packet, or if the link layer address of this node has changed.

ICMPv6 NS packets transmitted (icmpv6TxNs)

ICMPv6 neighbor solicitation packets transmitted. These packets are sent to get the link layer addresses of neighboring nodes or to confirm that they are reachable.

ICMPv6 RA packets transmitted (icmpv6TxRa)

ICMPv6 router advertisement packets transmitted. These packets are sent at regular intervals or in response to a router solicitation packet from a neighbor.

ICMPv6 RS packets transmitted (icmpv6TxRs)

ICMPv6 router solicitation packets transmitted. These packets are sent to request neighboring routers to generate router advertisements immediately rather than wait for the next defined time.

ICMPv6 Echo Request packets transmitted (icmpv6TxEchoReq)

ICMPv6 Ping Echo Request packets transmitted.

ICMPv6 Echo Reply packets transmitted (icmpv6TxEchoReply)

ICMP Ping Echo Reply packets transmitted.

ICMPv6 RA error packets (Error in RA packet)

ICMPv6 router advertisement error packets received that contain an error in the header, such as an incorrect source IP address, destination IP address, or packet length.

ICMPv6 NA error packets (Error in NA packet)

ICMPv6 neighbor advertisement error packets received that contain an error in the header, such as an incorrect source IP address, destination IP address, or packet length.

ICMPv6 NS error packets (Error in NS packet)

ICMPv6 neighbor solicitation error packets received that contain an error in the header, such as an incorrect source IP address, destination IP address, or packet length.

ICMPv6 bad checksum (Cksumerr)

Packets received with an ICMPv6 checksum error.

unsupported ICMPv6 packets (icmpv6Unspt)

ICMPv6 packets received that are not supported by the NetScaler.

Rate threshold exceeded packets (icmpv6thsld)

Packets dropped because the default threshold of 100 requests per 10 milliseconds has been exceeded.

This is a configurable value using the set rateControl command.

protocol ip

Sep 22, 2015

The following operations can be performed on "protocol ip":

Displays statistics of the IP protocol.

stat protocol ip [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

IP packets received (IPPktRx)

IP packets received.

IP bytes received (IPbRx)

Bytes of IP data received.

IP packets transmitted (IPPktTx)

IP packets transmitted.

IP bytes transmitted (IPbTx)

Bytes of IP data transmitted.

Megabits received (IPMbRx)

Megabits of IP data received.

Megabits transmitted (IPMbTx)

Megabits of IP data transmitted.

Total routed IP packets (IPRoutedPkts)

Total routed packets.

Total routed IP Mbits (IPRoutedMbits)

Total routed Mbits

IP fragments received (IPFragRx)

IP fragments received.

Successful reassembly (reasSucc)

Fragmented IP packets successfully reassembled on the NetScaler.

Reassembly attempted (reasAtmp)

IP packets that the NetScaler attempts to reassemble. If one of the fragments is missing, the whole packet is dropped.

IP address lookups (IpLkUp)

IP address lookups performed by the NetScaler. When a packet is received on a non-established session, the NetScaler checks if the destination IP address is one of the NetScaler owned IP addresses.

IP address lookup failure (IpLkFail)

IP address lookups performed by the NetScaler that have failed because the destination IP address of the packet does not match any of the NetScaler owned IP addresses.

UDP fragments forwarded (udpFgFwd)

UDP fragments forwarded to the client or the server.

TCP fragments forwarded (tcpFgFwd)

TCP fragments forwarded to the client or the server.

Fragmentation packets created (frgPktCr)

Fragmented packets created by the NetScaler.

Bad IP checksums (badCksum)

Packets received with an IP checksum error.

Unsuccessful reassembly (reasFail)

Packets received that could not be reassembled. This can occur when there is a checksum failure, an identification field mismatch, or when one of the fragments is missing.

Reassembled data too big (reasBig)

Packets received for which the reassembled data exceeds the Ethernet packet data length of 1500 bytes.

Zero fragment length received (zeroLen)

Packets received with a fragment length of 0 bytes.

Duplicate fragments received (dupFrag)

Duplicate IP fragments received. This can occur when the acknowledgement was not received within the expected

time.

Out of order fragment received (oooFrag)

Fragments received that are out of order.

Unknown destination received (UnkDst)

Packets received in which the destination IP address was not reachable or not owned by the NetScaler.

Bad Transport (badTran)

Packets received in which the protocol specified in the IP header is unknown to the NetScaler.

VIP down (vipDown)

Packets received for which the VIP is down. This can occur when all the services bound to the VIP are down or the VIP is manually disabled.

Fix header failure (hdrFail)

Packets received that contain an error in one or more components of the IP header.

TTL expired during transit (ttlExp)

Packets for which the time-to-live (TTL) expired during transit. These packets are dropped.

max non-TCP clients (maxClt)

Attempts to open a new connection to a service for which the maximum limit has been exceeded. Default value, 0, applies no limit.

Unknown services (UnkSvc)

Packets received on a port or service that is not configured.

land-attacks (LndAtk)

Land-attack packets received. The source and destination addresses are the same.

Invalid IP header size (errHdrSz)

Packets received in which an invalid data length is specified, or the value in the length field and the actual data length do not match. The range for the Ethernet packet data length is 0-1500 bytes.

Invalid IP packet size (errPktLen)

Total number of packets received by NetScaler with invalid IP packet size.

Truncated IP packet (trIP)

Truncated IP packets received. An overflow in the routers along the path can truncate IP packets.

Truncated non-IP packet (trNonIp)

Truncated non-IP packets received.

ZERO next hop (zrNxtHop)

Packets received that contain a 0 value in the next hop field. These packets are dropped.

Packets with len > 1514 rcvd (BadLenTx)

Packets received with a length greater than the normal maximum transmission unit of 1514 bytes.

Packets with bad MAC sent (BadMacTx)

IP packets transmitted with a bad MAC address.

protocol ipv6

Sep 22, 2015

The following operations can be performed on "protocol ipv6":

Displays statistics of the IPv6 protocol.

stat protocol ipv6 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

IPv6 packets received (ipv6RxPkts)

IPv6 packets received.

IPv6 bytes received (ipv6RxBytes)

Bytes of IPv6 data received.

IPv6 packets transmitted (ipv6TxPkts)

IPv6 packets transmitted

IPv6 bytes transmitted (ipv6TxBytes)

Bytes of IPv6 data transmitted.

Total Routed IPv6 packets (ipv6RoutePkts)

IPv6 packets routed.

Total Routed IPv6 Mbits (ipv6RouteMbits)

IPv6 total Mbits.

IPv6 Fragments received. (ipv6FragRxPkts)

IPv6 fragments received.

TCP Fragments reassembled. (ipv6FragTcpReass)

TCP fragments processed after reassembly.

UDP Fragments reassembled. (ipv6FragUdpReass)

UDP fragments processed after reassembly.

IPv6 Fragments processed without reassembly. (ipv6FragPktsProcessNoReass)

IPv6 fragments processed without reassembly.

IPv6 Fragments bridged. (ipv6FragPktsForward)

IPv6 fragments forwarded to the client or server without reassembly.

IPv6 error hdr packets (RxErrHdr)

Packets received that contain an error in one or more components of the IPv6 header.

IPv6 unsupported next header (Errnxthdr)

Packets received that contain an unsupported next header. The supported next headers are TCP, ICMP, UDP, OSPF, and FRAGMENT.

IPv6 Land-attacks (land attack)

Land-attack packets received. The source and destination addresses are the same. If not dropped, these packets can lock up the appliance.

Reassembled data too big (AssembledPktTooBig)

Packets received for which the reassembled data exceeds the Ethernet packet data length of 1500 bytes.

Zero fragment length received (ZeroLenFramentedPkt)

Packets received with a fragment length of 0 bytes.

ICMPv6 NA packets received

Number of ICMPv6 NA packets received by NetScaler (OBSOLETE).

ICMPv6 NS packets received

Number of ICMPv6 NS packets received by NetScaler (OBSOLETE).

ICMPv6 NA packets transmitted

Number of ICMPv6 NA packets transmitted by NetScaler (OBSOLETE).

ICMPv6 NS packets transmitted

Number of ICMPv6 NS packets transmitted by NetScaler (OBSOLETE).

ICMPv6 RA packets received

Number of ICMPv6 RA packets received by NetScaler (OBSOLETE).

ICMPv6 RS packets transmitted

Number of ICMPv6 RS packets transmitted by NetScaler (OBSOLETE).

ICMPv6 packets received

Number of ICMPv6 packets received by NetScaler (OBSOLETE).

ICMPv6 packets transmitted

Number of ICMPv6 packets transmitted by NetScaler (OBSOLETE).

IPv6 error hdr packets

Number of erroneous header packets received (OBSOLETE).

IPv6 error packets

Number of erroneous packets received (OBSOLETE).

IPv6 bad checksum

Number of bad checksum packets received (OBSOLETE).

ICMPv6 error packets

Number of erroneous ICMPv6 packets received (OBSOLETE).

unsupported ICMPv6 packets

Number of ICMPv6 unsupported packets received (OBSOLETE).

Rate threshold exceeded packets

Number of ICMPv6 packets dropped for rate threshold exceeded (OBSOLETE).

protocol tcp

Sep 22, 2015

The following operations can be performed on "protocol tcp":

Displays statistics of the TCP protocol.

stat protocol tcp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Server active connections (ActSvrCo)

Connections to a server currently responding to requests.

Opening server connections (SvrCxO)

Server connections in the Opening state, which indicates that the handshakes are not yet complete.

Opening client connections (CltCxO)

Client connections in the Opening state, which indicates that the handshakes are not yet complete.

Established client connections (CltCxE)

Current client connections in the Established state, which indicates that data transfer can occur between the NetScaler and the client.

Established server connections (SvrCxE)

Current server connections in the Established state, which indicates that data transfer can occur between the NetScaler and the server.

TCP packets received (TCPPktRx)

TCP packets received.

TCP bytes received (TCPbRx)

Bytes of TCP data received.

TCP packets transmitted (TCPPktTx)

TCP packets transmitted.

TCP bytes transmitted (TCPbTx)

Bytes of TCP data transmitted.

All client connections (CltCx)

Client connections, including connections in the Opening, Established, and Closing state.

Closing client connections (CltCxCl)

Client connections in the Closing state, which indicates that the connection termination process has initiated but is not complete.

Opened client connections (TotCltO)

Client connections opened by the NetScaler since startup (after three-way handshake). This counter is reset when the NetScaler is restarted.

All server connections (SvrCx)

Server connections, including connections in the Opening, Established, and Closing state.

Closing server connections (SvrCxCl)

Server connections in the Closing state, which indicates that the connection termination process has initiated but is not complete.

Opened server connections (TotSvrO)

Server connections initiated by the NetScaler since startup. This counter is reset when the NetScaler is restarted.

Surge queue (SQlen)

Connections in the surge queue. When the NetScaler cannot open a connection to the server, for example when maximum connections have been reached, the NetScaler queues these requests.

Spare connections (SpConn)

Spare connections available. To save time and resources in establishing another connection for a new client, the connection on the server is not closed after completing the request from the first client and is available for serving future requests.

Client idle flushed (ZomCltF)

Client connections that are flushed because the client has been idle for some time.

Client half opened flushed (ZCltFHo)

Half-opened client connections that are flushed because the three-way handshakes are not complete.

Client active half closed flushed (ZCltFAhc)

Active half-closed client connections that are flushed because the client has closed the connection and there has

been no activity on the connection.

Client passive half closed flushed (ZCltFPhc)

Passive half-closed client connections that are flushed because the NetScaler has closed the connection and there has been no activity on the connection.

Server idle connections flushed (ZSvrF)

Server connections that are flushed because there have been no client requests in the queue for some time.

Server half opened flushed (ZSvrFHo)

Half-opened server connections that are flushed because the three-way handshakes are not complete.

Server active half closed flushed (ZSvrFAhc)

Active half-closed server connections that are flushed because the server has closed the connection and there has been no activity on the connection.

Server passive half closed flushed (ZSrvFPhc)

Passive half-closed server connections that are flushed because the NetScaler has closed the connection and there has been no activity on the connection.

Zombie cleanup calls (ZmbCall)

Times the Zombie cleanup function is called. Every time a connection is flushed, it is marked for cleanup. The Zombie cleanup function clears all these connections at predefined intervals.

SYN packets received (TCPSYN)

SYN packets received

Server probes (SYNProbe)

Probes from the NetScaler to a server. The NetScaler sends a SYN packet to the server to check its availability and expects a SYN_ACK packet from the server before a specified response timeout.

FIN packets from server (SvrFin)

FIN packets received from the server.

FIN packets from client (CltFin)

FIN packets received from the clients.

Time wait to SYN (WaToSyn)

SYN packets (packets used to initiate a TCP connection) received on connections that are in the TIME_WAIT state. Packets cannot be transferred on a connection in this state.

Data in TIME_WAIT (WaDat)

Bytes of data received on connections that are in the TIME_WAIT state. Data cannot be transferred on a

connection that is in this state.

SYN packets held (SYNHeld)

SYN packets held on the NetScaler that are waiting for a server connection.

SYN packets flushed (SYNFlush)

SYN packets flushed on the NetScaler because of no response from the server for three or more seconds.

TIME_WAIT connections closed (FinWaitC)

Connections closed on the NetScaler because the number of connections in the TIME_WAIT state has exceeded the default value of 7000.

Bad TCP checksum (TCPBadCk)

Packets received with a TCP checksum error.

Data after FIN (TCPDtFin)

Packets received following a connection termination request. This error is usually caused by a reordering of packets during transmission.

SYN in SYN_RCVD state (TCPSYNRv)

SYN packets received on a connection that is in the SYN_RCVD state. A connection goes into the SYN_RCVD state after receiving a SYN packet.

SYN in ESTABLISHED state (TCPSYNEs)

SYN packets received on a connection that is in the ESTABLISHED state. A SYN packet is not expected on an ESTABLISHED connection.

SYN_SENT incorrect ACK packet (TCPBadAk)

Incorrect ACK packets received on a connection that is in the SYN_SENT state. An incorrect ACK packet is the third packet in the three-way handshake that has an incorrect sequence number.

RST packets received (TCPRST)

Reset packets received from a client or a server.

RST on not ESTABLISHED (TCPRSTNE)

Reset packets received on a connection that is not in the ESTABLISHED state.

RST out of window (TCPRSTOW)

Reset packets received on a connection that is out of the current TCP window.

RST in TIME_WAIT (TCPRSTTi)

Reset packets received on a connection that is in the TIME_WAIT state. Packets cannot be transferred on a connection in the TIME_WAIT state.

Server out of order packets (SvrOOO)

Out of order TCP packets received from a server.

Client out of order packets (CltOOO)

Out of order TCP packets received from a client.

TCP hole on client connection (CltHole)

TCP holes created on a client connection. When out of order packets are received from a client, a hole is created on the NetScaler for each group of missing packets.

TCP hole on server connection (SvrHole)

TCP holes created on a server connection. When out of order packets are received from a server, a hole is created on the NetScaler for each group of missing packets.

Seq number SYN cookie reject (CSeqRej)

SYN cookie packets rejected because they contain an incorrect sequence number.

Signature SYN cookie reject (CSigRej)

SYN cookie packets rejected because they contain an incorrect signature.

Seq number SYN cookie drop (CSigDrp)

SYN cookie packets dropped because the sequence number specified in the packets is outside the current window.

MSS SYN cookie reject (CMssRej)

SYN cookie packets rejected because the maximum segment size (MSS) specified in the packets is incorrect.

Any IP port allocation failure (PortFal)

Port allocations that have failed on a mapped IP address because the maximum limit of 65536 has been exceeded.

IP port allocation failure (PortFall)

Port allocations that have failed on a subnet IP address or vserver IP address because the maximum limit of 65536 has been exceeded.

Stray packets (StrayPkt)

Number of stray or misrouted packets.

RST packets sent (SentRst)

Reset packets sent to a client or a server.

Bad state connections (BadConn)

Connections that are not in a valid TCP state.

RST threshold dropped (RstThre)

Reset packets dropped because the default threshold of 100 resets per 10 milliseconds has been exceeded. This is a configurable value using the set rateControl command.

Packets out of window (OOWPkt)

Packets received that are out of the current advertised window.

SYNs dropped (Congestion) (SynCng)

SYN packets dropped because of network congestion.

Client retransmissions (TCPCltRe)

Packets retransmitted by a client. This usually occurs because the acknowledgement from the NetScaler has not reached the client.

Full packet retransmissions (TCPFulRe)

Full packets retransmitted by the client or the server.

SYN packet retries (TCPSYNRe)

SYN packets resent to a server.

SYN packets timeout (TCPSYNG)

Attempts to establish a connection on the NetScaler that timed out.

TCP retransmission (Retr)

TCP packets retransmitted. The NetScaler attempts to retransmit the packet up to seven times, after which it resets the other half of the TCP connection.

1st retransmission (1stRetr)

Packets retransmitted once by the NetScaler.

3rd retransmission (3rdRetr)

Packets retransmitted three times by the NetScaler.

5th retransmission (5thRetr)

Packets retransmitted five times by the NetScaler.

7th retransmission (7thRetr)

Packets retransmitted seven times by the NetScaler. If this fails, the NetScaler terminates the connection.

Fast retransmits (FastRetr)

TCP packets on which the NetScaler performs a fast retransmission in response to three duplicate acknowledgements or a partial acknowledgement. The NetScaler assumes that the packet is lost and retransmits the packet before its time-out.

Server retransmissions (TCPSvrRe)

Packets retransmitted by a server. This usually occurs because the acknowledgement from the NetScaler has not reached the server.

Partial packet retransmissions (TCPParRe)

Partial packet retransmits by a client or server due to congestion on the connection. This usually occurs because the window advertised by the NetScaler is not big enough to hold the full packet.

FIN packet retries (TCPFINRe)

FIN packets resent to a server or a client.

FIN packets timeout (TCPFING)

Connections that were timed out by the NetScaler because of not receiving the ACK packet after retransmitting the FIN packet four times.

2nd retransmission (2ndRetr)

Packets retransmitted twice by the NetScaler.

4th retransmission (4thRetr)

Packets retransmitted four times by the NetScaler.

6th retransmission (6thRetr)

Packets retransmitted six times by the NetScaler.

TCP retransmission giveup (RetrG)

Number of times NetScaler terminates a connection after retransmitting the packet seven times on that connection.Retrasnmission happens when recieving end doesn't acknowledges the packet.

TCP level cip failure (CltHdrEr)

Number of times TCP level client header insertion failure

protocol udp

Sep 22, 2015

The following operations can be performed on "protocol udp":

Displays statistics of the UDP protocol.

stat protocol udp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Packets received (UDPPktRx)

Total number of UDP packets received.

Bytes received (UDPbRx)

Total number of UDP data received in bytes.

Packets transmitted (UDPPktTx)

Total number of UDP packets transmitted.

Bytes transmitted (UDPbTx)

Total number of UDP data transmitted in bytes.

Current rate threshold (UDPThs)

Limit for UDP packets handled every 10 milliseconds. Default value, 0, applies no limit.

This is a configurable value using the set rateControl command.

Unknown service (UDPUnSvc)

Stray UDP packets dropped due to no configured listening service.

Bad UDP checksum (UDPBadCkSum)

Packets received with a UDP checksum error.

Rate threshold exceeded (UDPRtEx)

Number of times the UDP rate threshold is exceeded. If this counter continuously increases, first make sure the UDP packets received are genuine.

If they are, increase the current rate threshold. This is a configurable value using the set rateControl command.

QOS Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- qos
- qos stats

qos

Sep 22, 2015

The following operations can be performed on "qos":

Display QoS statistics.

stat qos [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

IPC messages sent from QoS (ipcsent)

IPC messages sent from qos system.

IPC messages QoS failed to send (ipcfailed)

IPC messages failed to send from qos system.

IPC messages QoS received (ipcrcvd)

IPC messages received by qos.

IPC messages sent to QoS (pe2qsent)

IPC messages sent to qos system.

IPC messages failed to send QoS (pe2qfail)

IPC messages failed to send to qos system.

IPC messages received from QoS (pe2qrecv)

IPC messages received from qos system.

Bytes QoS marked for drop (bytsdrop)

Bytes QoS marked for drop

QoS bytes sent not classified (bytsntnc)

Bytes scheduled by QoS that were not classified.

QoS bytes dropped no connection (bytdrpnc)

Bytes dropped by QoS when no connection was found.

Packets sent to QoS (qosinpkt)

Packets sent to QoS for scheduling.

Packets from QoS to be sent (qosotpkt)

Packets from QoS to be sent

Packets Dropped by QoS (qosdrpkt)

Packets Dropped by QoS.

Classified source MAC rewritten (qosrwmac)

Number of packets with inband classification in source MAC.

QoS packets unclassified (qosuclas)

Number of packets without classification.

QoS packets classified (qosclas)

Number of packets with classification.

QoS learned true MAC (qoslm)

QoS learned true MAC

QoS Input Bytes (qosib)

Bytes sent to QoS for scheduling

QoS Output Bytes (qosob)

Bytes received from QoS to be sent

QoS Free Held List (qosfc)

No. more packets QoS can hold onto.

QoS Link 00 Bytes Sent (qosl00sd)

QoS bytes sent on Link 00

QoS Link 00 Bytes Dropped (qosl00dr)

QoS bytes dropped on Link 00

QoS Link 01 Bytes Sent (qosl01sd)

QoS bytes sent on Link 01

QoS Link 01 Bytes Dropped (qosl01dr)

QoS bytes dropped on Link 01

QoS Link 02 Bytes Sent (qosl02sd)

QoS bytes sent on Link 02

QoS Link 02 Bytes Dropped (qosl02dr)

QoS bytes dropped on Link 02

QoS Link 03 Bytes Sent (qosl03sd)

QoS bytes sent on Link 03

QoS Link 03 Bytes Dropped (qosl03dr)

QoS bytes dropped on Link 03

QoS Link 04 Bytes Sent (qosl04sd)

QoS bytes sent on Link 04

QoS Link 04 Bytes Dropped (qosl04dr)

QoS bytes dropped on Link 04

QoS Link 05 Bytes Sent (qosl05sd)

QoS bytes sent on Link 05

QoS Link 05 Bytes Dropped (qosl05dr)

QoS bytes dropped on Link 05

QoS Link 06 Bytes Sent (qosl06sd)

QoS bytes sent on Link 06

QoS Link 06 Bytes Dropped (qosl06dr)

QoS bytes dropped on Link 06

QoS Link 07 Bytes Sent (qosl07sd)

QoS bytes sent on Link 07

QoS Link 07 Bytes Dropped (qosl07dr)

QoS bytes dropped on Link 07

QoS Link 08 Bytes Sent (qosl08sd)

QoS bytes sent on Link 08

QoS Link 08 Bytes Dropped (qosl08dr)

QoS bytes dropped on Link 08

QoS Link 09 Bytes Sent (qosl09sd)

QoS bytes sent on Link 09

QoS Link 09 Bytes Dropped (qosl09dr)

QoS bytes dropped on Link 09

QoS Link 10 Bytes Sent (qosl10sd)

QoS bytes sent on Link 10

QoS Link 10 Bytes Dropped (qosl10dr)

QoS bytes dropped on Link 10

qos stats

Sep 22, 2015

The following operations can be performed on "qos stats":

show qos stats is an alias for stat qos

show qos stats - alias for 'stat qos'

Rewrite Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- rewrite action
- rewrite global
- rewrite param
- rewrite policy
- rewrite policylabel

rewrite action

Sep 22, 2015

The following operations can be performed on "rewrite action":

add | rm | set | unset | show | rename

Creates a rewrite action, which specifies exactly what modifications to make to a request or response before forwarding that request or response to the protected web server or to the user. In addition to user-defined actions, the rewrite feature has the following three built-in actions: * NOREWRITE - Sends the request or response to the user without rewriting it. * RESET - Resets the connection and notifies the user?s browser, so that the user can resend the request. * DROP - Drops the connection without sending a response to the user. One of the following three flow types is implicitly associated with every action: * Request - Action applies to the request. * Response - Action applies to the response. * Neutral - Action applies to both requests and responses.

add rewrite action <name> <type> <target> [<stringBuilderExpr>] [-pattern <expression> | -search <expression>] [-bypassSafetyCheck (YES | NO)] [-refineSearch <string>] [-comment <string>]

name

Name for the user-defined rewrite action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite action" or ?my rewrite action?).

type

Type of user-defined rewrite action. The information that you provide for, and the effect of, each type are as follows::

* REPLACE <target> <string_builder_expr>. Replaces the string with the string-builder expression.

* REPLACE_ALL <target> <string_builder_expr1> -(pattern|search) <string_builder_expr2>. In the request or response specified by <target>, replaces all occurrences of the string defined by <string_builder_expr1> with the string defined by <string_builder_expr2>. You can use a PCRE-format pattern or the search facility to find the strings to be replaced.

* REPLACE_HTTP_RES <string_builder_expr>. Replaces the complete HTTP response with the string defined by the string-builder expression.

* REPLACE_SIP_RES <target> - Replaces the complete SIP response with the string specified by <target>.

* INSERT_HTTP_HEADER <header_string_builder_expr> <contents_string_builder_expr>. Inserts the HTTP header specified by <header_string_builder_expr> and header contents specified by <contents_string_builder_expr>.

* DELETE_HTTP_HEADER <target>. Deletes the HTTP header specified by <target>.

* CORRUPT_HTTP_HEADER <target>. Replaces the header name of all occurrences of the HTTP header specified by <target> with a corrupted name, so that it will not be recognized by the receiver Example: MY_HEADER is changed to MHEY_ADER.

* INSERT_BEFORE <string_builder_expr1> <string_builder_expr1>. Finds the string specified in <string_builder_expr1> and inserts the string in <string_builder_expr2> before it.

* INSERT_BEFORE_ALL <target> <string_builder_expr1> -(pattern|search) <string_builder_expr2>. In the request or response specified by <target>, locates all occurrences of the string specified in <string_builder_expr1> and inserts the string specified in <string_builder_expr2> before each. You can use a PCRE-format pattern or the search facility to find the strings.

* INSERT_AFTER <string_builder_expr1> <string_builder_expr2>. Finds the string specified in <string_builder_expr1>, and inserts the string specified in <string_builder_expr2> after it.

* INSERT_AFTER_ALL <target> <string_builder_expr1> -(pattern|search) <string_builder_expr>. In the request or response specified by <target>, locates all occurrences of the string specified by <string_builder_expr1> and inserts the string specified by <string_builder_expr2> after each. You can use a PCRE-format pattern or the search facility to find the strings.

* DELETE <target>. Finds and deletes the specified target.

* DELETE_ALL <target> -(pattern|search) <string_builder_expr>. In the request or response specified by <target>, locates and deletes all occurrences of the string specified by <string_builder_expr>. You can use a PCRE-format pattern or the search facility to find the strings.

Possible values: noop, delete, insert_http_header, delete_http_header, corrupt_http_header, insert_before, insert_after, replace, replace_http_res, delete_all, replace_all, insert_before_all, insert_after_all, clientless_vpn_encode, clientless_vpn_encode_all, clientless_vpn_decode, clientless_vpn_decode_all, insert_sip_header, delete_sip_header, corrupt_sip_header, replace_sip_res

target

Default syntax expression that specifies which part of the request or response to rewrite.

stringBuilderExpr

Default syntax expression that specifies the content to insert into the request or response at the specified location, or that replaces the specified string.

pattern

Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (_) and space ( ) that is not otherwise used in the expression. Example: re~https?://|HTTPS?://~ The preceding regular expression can use the tilde (~) as the delimiter because that character does not appear in the regular expression itself. Used in the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

search

Search facility that is used to match multiple strings in the request or response. Used in the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types. The following search types are supported:

* Text ("text(string)") - A literal string. Example: -search text("hello")

* Regular expression (?regex(re<delimiter>regular exp<delimiter>)?) - Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (_) and space ( ) that is not otherwise used in the expression. Example: -search regex(re~^hello~) The preceding regular expression can use the tilde (~) as the delimiter because that character does not appear in the regular expression itself.

* XPath ("xpath(xp<delimiter>xpath expression<delimiter>)") - An XPath expression. Example: -search xpath(xp%/a/b%)

* JSON ("xpath_json(xp<delimiter>xpath expression<delimiter>)") - An XPath JSON expression. Example: -search xpath_json(xp%/a/b%)

NOTE: JSON searches use the same syntax as XPath searches, but operate on JSON files instead of standard XML files.

* Patset ("patset(patset)") - A predefined pattern set. Example: -search patset("patset1")

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions. An unsafe expression is one that contains references to message elements that might not be present in all messages. If an expression refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

refineSearch

Specify additional criteria to refine the results of the search.

Always starts with the "extend(m,n)" operation, where 'm' specifies number of bytes to the left of selected data and 'n' specifies number of bytes to the right of selected data.

You can use refineSearch only on body expressions, and for the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

comment

Comment. Can be used to preserve information about this rewrite action.

i) add rewrite action act_insert INSERT_HTTP_HEADER change_req "\\\\"no change\\\\"" . This Adds to http header will add the header change_req: no change. ii) add rewri

---

Removes a rewrite action.

rm rewrite action <name>

name

Name of the rewrite action to remove.

rm rewrite action act_before

Modifies the specified parameters of a rewrite action.

set rewrite action <name> [-target <string>] [-stringBuilderExpr <string>] [-pattern <expression> | -search <expression>] [-bypassSafetyCheck (YES | NO)] [-refineSearch <string>] [-comment <string>]

name

Name of the rewrite action to modify.

target

Expression that specifies which part of the connection to rewrite.

stringBuilderExpr

Default syntax expression that specifies the content to insert into the request or response at the specified location, or that replaces the specified string.

pattern

Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (_) and space ( ) that is not otherwise used in the expression. Example: re~https?://|HTTPS?://~ The preceding regular expression can use the tilde (~) as the delimiter because that character does not appear in the regular expression itself. Used in the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

search

Search facility that is used to match multiple strings in the request or response. Used in the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types. The following search types are supported:

* Text ("text(string)") - A literal string. Example: -search text("hello")

* Regular expression (?regex(re<delimiter>regular exp<delimiter>)?) - Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (_) and space ( ) that is not otherwise used in the expression. Example: -search regex(re~^hello~) The preceding regular expression can use the tilde (~) as the delimiter because that character does not appear in the regular expression itself.

* XPath ("xpath(xp<delimiter>xpath expression<delimiter>)") - An XPath expression. Example: -search xpath(xp%/a/b%)

* JSON ("xpath_json(xp<delimiter>xpath expression<delimiter>)") - An XPath JSON expression. Example: -search xpath_json(xp%/a/b%)

NOTE: JSON searches use the same syntax as XPath searches, but operate on JSON files instead of standard XML files.

* Patset ("patset(patset)") - A predefined pattern set. Example: -search patset("patset1")

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions. An unsafe expression is one that contains references to message elements that might not be present in all messages. If an expression refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

refineSearch

Specify additional criteria to refine the results of the search.

Always starts with the "extend(m,n)" operation, where 'm' specifies number of bytes to the left of selected data and 'n' specifies number of bytes to the right of selected data.

You can use refineSearch only on body expressions, and for the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

comment

Comment. Can be used to preserve information about this rewrite action.

```
set rewrite action rwact1 -target "HTTP.REQ.HEADER(\\\"MyHdr\\\")" -stringBuilderExpr "HTTP.REQ.URL.MARK_SAFE"
```

Use this command to remove rewrite action settings.Refer to the set rewrite action command for meanings of the arguments.

```
unset rewrite action <name> [-stringBuilderExpr] [-refineSearch] [-comment]
```

Displays the current settings for the specified rewrite action. If no rewrite action name is provided, displays a list of all rewrite actions currently configured on the NetScaler appliance.

```
show rewrite action [<name>]
```

name

Name of the rewrite action.

summary

fullValues

format

level

stateflag

type

Type of rewrite action. It can be: (delete | replace | insert_http_header | insert_before | insert_after | replace_http_res).

target

Expression specifying which part of HTTP header needs to be rewritten.

stringBuilderExpr

Expression specifying the value of rewritten HTTP header.

pattern

Pattern used for insert_before_all, insert_after_all, replace_all, delete_all action types.

search

Search facility that is used to match multiple strings in the request or response. Used in the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types. The following search types are supported:

* Text ("text(string)") - A literal string. Example: -search text("hello")

* Regular expression (?regex(re<delimiter>regular exp<delimiter>)?) - Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (_) and space ( ) that is not otherwise used in the expression. Example: -search regex(re~^hello~) The preceding regular expression can use the tilde (~) as the delimiter because that character does not appear in the regular expression itself.

* XPath ("xpath(xp<delimiter>xpath expression<delimiter>)") - An XPath expression. Example: -search xpath(xp%/a/b%)

* JSON ("xpath_json(xp<delimiter>xpath expression<delimiter>)") - An XPath JSON expression. Example: -search xpath_json(xp%/a/b%)

NOTE: JSON searches use the same syntax as XPath searches, but operate on JSON files instead of standard XML files.

* Patset ("patset(patset)") - A predefined pattern set. Example: -search patset("patset1")

bypassSafetyCheck

The safety check to allow unsafe expressions.

refineSearch

Specify additional criteria to refine the results of the search.

Always starts with the "extend(m,n)" operation, where 'm' specifies number of bytes to the left of selected data and 'n' specifies number of bytes to the right of selected data.

You can use refineSearch only on body expressions, and for the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

hits

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

description

Description of the action

flags

isDefault

A value of true is returned if it is a default rewriteaction.

comment

Comment. Can be used to preserve information about this rewrite action.

builtin

Flag to determine whether rewrite action is built-in or not

devno

count

1. show rewrite action 2. show rewrite action act_insert

Renames a rewrite action.

rename rewrite action <name>@ <newName>@

name

Existing name of the rewrite action.

newName

New name for the rewrite action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite action" or ?my rewrite action?).

rename rewrite action oldname newname

rewrite global

Sep 22, 2015

The following operations can be performed on "rewrite global":

bind | unbind | show

Activates the specified rewrite policy globally.

bind rewrite global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]

policyName

Name of the rewrite policy to activate.

i) bind rewrite global pol9 9 ii) bind rewrite global pol9 9 120 iii) bind rewrite global pol9 9 "HTTP.REQ.HEADER(\\\\"qh3\\\\").TYPECAST_NUM_T(DECIMAL)"

Unbinds the specified rewrite policy from rewrite global. See the bind rewrite global command for a description of the parameters.

unbind rewrite global <policyName> [-type <type>] [-priority <positive_integer>]

policyName

Name of the rewrite policy to deactivate.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind rewrite global pol9

Displays the list of policies bound to the specified rewrite global policy bank. If no policy bank is specified, displays a list of all policies bound to rewrite global.

show rewrite global [-type <type>]

type

The bindpoint to which to policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, OTHERTCP_RES_OVERRIDE, OTHERTCP_RES_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, SIPUDP_RES_OVERRIDE, SIPUDP_RES_DEFAULT

summary

fullValues

format

level

stateflag

policyName

Name of the rewrite policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

* reqvserver - Forward the request to the specified request virtual server.

* resvserver - Forward the response to the specified response virtual server.

* policylabel - Invoke the specified policy label.

labelName

* If labelType is policylabel, name of the policy label to invoke.

* If labelType is reqvserver or resvserver, name of the virtual server to which to forward the request of response.

numpol

The number of policies bound to the bindpoint.

flowType

flowtype of the bound rewrite policy.

flags

devno

count

show rewrite global

rewrite param

Sep 22, 2015

The following operations can be performed on "rewrite param":

set | unset | show

Sets the default rewrite undefined action. If an UNDEF event is triggered during policy evaluation and if no undefAction is specified for the current policy, this value is used.

set rewrite param -undefAction <string>

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition.

Available settings function as follows:

* NOOP - Send the request to the protected server instead of responding to it.

* RESET - Reset the request and notify the user?s browser, so that the user can resend the request.

* DROP - Drop the request without sending a response to the user.

Default value: "NOREWRITE"

set rewrite param -undefAction RESET

Resets the global undefAction to NOREWRITE..Refer to the set rewrite param command for meanings of the arguments.

unset rewrite param -undefAction

unset rewrite param -undefAction

Displays the default rewrite undefAction.

show rewrite param

format

level

undefAction

Name of the rewrite action.

show rewrite param

rewrite policy

Sep 22, 2015

The following operations can be performed on "rewrite policy":

add | rm | set | unset | show | stat | rename

Creates a rewrite policy, which specifies which requests or responses to rewrite.

add rewrite policy <name> <rule> <action> [<undefAction>] [-comment <string>] [-logAction <string>]

name

Name for the rewrite policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy" or ?my rewrite policy?).

rule

Expression against which traffic is evaluated. Written in default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the rewrite action to perform if the request or response matches this rewrite policy.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition.

Available settings function as follows:

* NOREWRITE - Send the request to the protected server without rewriting.

* RESET - Reset the request and notify the user?s browser, so that the user can resend the request.

* DROP - Drop the request without sending a response to the user.

comment

Any comments to preserve information about this rewrite policy.

logAction

Name of messagelog action to use when a request matches this policy.

i) add rewrite policy pol9 "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh3\\\")" act_insert ii) add rewrite policy pol9 "HTTP.REQ.HEADER(\\\"header\\\").CONTAIN

Removes the specified rewrite policy.

rm rewrite policy <name>

name

Name of the rewrite policy to be removed.

rm rewrite policy pol9

Modifies the specified parameters of a rewrite policy.

set rewrite policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>] [-logAction <string>]

name

Name of the rewrite policy to modify.

rule

Expression against which traffic is evaluated. Written in default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the rewrite action to perform if the request or response matches this rewrite policy.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition.

Available settings function as follows:

* NOREWRITE - Send the request to the protected server without rewriting.

* RESET - Reset the request and notify the user?s browser, so that the user can resend the request.

* DROP - Drop the request without sending a response to the user.

comment

Any comments to preserve information about this rewrite policy.

logAction

Name of messagelog action to use when a request matches this policy.

set rewrite policy pol9 -rule "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"

Removes the settings of an existing rewrite policy. Attributes for which a default value is available revert to their default values. See the set rewrite policy command for a description of the parameters..Refer to the set rewrite policy command for meanings of the arguments.

unset rewrite policy <name> [-undefAction] [-comment] [-logAction]

unset rewrite policy pol9 -undefAction

Displays the current settings for the specified rewrite policy. If no policy name is provided, displays a list of all rewrite policies currently configured on the NetScaler appliance.

show rewrite policy [<name>] show rewrite policy stats - alias for 'stat rewrite policy'

name

Name of the rewrite policy.

summary

fullValues

format

level

stateflag

rule

Expression against which traffic is evaluated. Written in default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Rewrite action associated with the policy.

undefAction

Undef Action associated with the policy.

hits

Number of hits.

undefHits

Number of Undef hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

comment

Any comments to preserve information about this rewrite policy.

logAction

Name of messagelog action to use when a request matches this policy.

bindPolicyType

isDefault

A value of true is returned if it is a default rewritepolicy.

vserverType

builtin

Flag to determine if rewrite policy is built-in or not

devno

count

show rewrite policy

Displays statistics for the specified rewrite policy. If no policy name is specified, displays abbreviated statistics for all rewrite policies currently configured on the NetScaler appliance.

stat rewrite policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the rewrite policy.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

stat rewrite policy

Renames the specified rewrite policy. You must restart the NetScaler appliance to put new name in effect.

rename rewrite policy <name>@ <newName>@

name

Existing name of the rewrite policy.

newName

New name for the rewrite policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy" or ?my rewrite policy?).

rename rewrite policy oldname newname

rewrite policylabel

Sep 22, 2015

The following operations can be performed on "rewrite policylabel":

add | rm | bind | unbind | show | stat | rename

Creates a user-defined rewrite policy label.

add rewrite policylabel <labelName> <transform>

labelName

Name for the rewrite policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rewrite policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy label" or ?my rewrite policy label?).

transform

Types of transformations allowed by the policies bound to the label. For Rewrite, the following types are supported:

* http_req - HTTP requests

* http_res - HTTP responses

* othertcp_req - Non-HTTP TCP requests

* othertcp_res - Non-HTTP TCP responses

* url - URLs

* text - Text strings

* clientless_vpn_req - NetScaler clientless VPN requests

* clientless_vpn_res - NetScaler clientless VPN responses

* sipudp_req - SIP requests

* sipudp_res - SIP responses

Possible values: http_req, http_res, othertcp_req, othertcp_res, url, text, clientless_vpn_req, clientless_vpn_res, sipudp_req, sipudp_res

add rewrite policylabel trans_http_url http_req

Removes the specified rewrite policy label.

rm rewrite policylabel <labelName>

labelName

Name of the rewrite policy label to remove.

rm rewrite policylabel trans_http_url

Binds the specified rewrite policy to the specified policy label.

bind rewrite policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

labelName

Name of the rewrite policy label to which to bind the policy.

policyName

Name of the rewrite policy to bind to the policy label.

i) bind rewrite policylabel trans_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind rewrite policylabel trans_http_url pol_2 2

Unbinds the specified rewrite policy from the specified policy label. See the bind rewrite policylabel command for a description of the parameters.

unbind rewrite policylabel <labelName> <policyName> [-priority <positive_integer>]

labelName

Name for the rewrite policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rewrite policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy label" or ?my rewrite policy label?).

policyName

Name of the rewrite policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind rewrite policylabel trans_http_url pol_1

Displays the current settings for the specified rewrite policy label. If no policy label is specified, displays a list of all rewrite policy labels currently configured on the NetScaler appliance.

show rewrite policylabel [<labelName>]

labelName

Name of the rewrite policy label.

summary

fullValues

format

level

stateflag

transform

Types of transformations allowed by the policies bound to the label. For Rewrite, the following types are supported:

* http_req - HTTP requests

* http_res - HTTP responses

* othertcp_req - Non-HTTP TCP requests

* othertcp_res - Non-HTTP TCP responses

* url - URLs

* text - Text strings

* clientless_vpn_req - NetScaler clientless VPN requests

* clientless_vpn_res - NetScaler clientless VPN responses

* sipudp_req - SIP requests

* sipudp_res - SIP responses

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the rewrite policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Suspend evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

* reqvserver - Forward the request to the specified request virtual server.

* resvserver - Forward the response to the specified response virtual server.

* policylabel - Invoke the specified policy label.

labelName

* If labelType is policylabel, name of the policy label to invoke.

* If labelType is reqvserver or resvserver, name of the virtual server to which to forward the request or response.

flowType

Flowtype of the bound rewrite policy.

description

Description of the policylabel

isDefault

A value of true is returned if it is a default rewritepolicylabel.

flags

devno

count

i) show rewrite policylabel trans_http_url ii) show rewrite policylabel

---

Displays statistics for the specified rewrite policy label. If no policy label name is provided, displays abbreviated statistics for all rewrite policy labels currently configured on the NetScaler appliance.

stat rewrite policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

labelName

Name of the rewrite policy label.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy Label Hits (Hits)

Number of times policy label was invoked.

Renames a rewrite policy label.

rename rewrite policylabel <labelName>@ <newName>@

labelName

Current name of the policy label.

newName

New name for the rewrite policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy label" or ?my policy label?).

rename rewrite policylabel oldname newname

Responder Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- responder action
- responder global
- responder htmlpage
- responder param
- responder policy
- responder policylabel

responder action

Sep 22, 2015

The following operations can be performed on "responder action":

add | rm | set | unset | show | rename

Creates a responder action, which specifies how to respond to a request.

add responder action <name> <type> (<target> | <htmlpage>) [-bypassSafetyCheck (YES | NO)] [-comment <string>]

name

Name for the responder action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the responder policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder action" or 'my responder action').

type

Type of responder action. Available settings function as follows:

* respondwith <target> - Respond to the request with the expression specified as the target.

* respondwithhtmlpage - Respond to the request with the uploaded HTML page object specified as the target.

* redirect - Redirect the request to the URL specified as the target.

* sqlresponse_ok - Send an SQL OK response.

* sqlresponse_error - Send an SQL ERROR response.

Possible values: noop, respondwith, redirect, respondwithhtmlpage, sqlresponse_ok, sqlresponse_error

target

Expression specifying what to respond with. Typically a URL for redirect policies or a default-syntax expression. In addition to NetScaler default-syntax expressions that refer to information in the request, a stringbuilder expression can contain text and HTML, and simple escape codes that define new lines and paragraphs. Enclose each stringbuilder expression element (either a NetScaler default-syntax expression or a string) in double quotation marks. Use the plus (+) character to join the elements.

Examples:

1) Respondwith expression that sends an HTTP 1.1 200 OK response:

"HTTP/1.1 200 OK\\r\\n\\r\\n"

2) Redirect expression that redirects user to the specified web host and appends the request URL to the redirect.

"http://backupsite2.com" + HTTP.REQ.URL

3) Respondwith expression that sends an HTTP 1.1 404 Not Found response with the request URL included in the response:

"HTTP/1.1 404 Not Found\\r\\n\\r\\n"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."

The following requirement applies only to the NetScaler CLI:

Enclose the entire expression in single quotation marks. (NetScaler default expression elements should be included inside the single quotation marks for the entire expression, but do not need to be enclosed in double quotation marks.)

htmlpage

For respondwithhtmlpage policies, name of the HTML page object to use as the response. You must first import the page object.

bypassSafetyCheck

Bypass the safety check, allowing potentially unsafe expressions. An unsafe expression in a response is one that contains references to request elements that might not be present in all requests. If a response refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

comment

Comment. Any type of information about this responder action.

1) add responder action act1 respondwith "\\\"HTTP/1.1 200 OK\\\\r\\\\n\\\\r\\\\n\\\"" 2) add responder action resp respondwithhtmlpage my-responder-page, 3) add responde

Removes the specified responder action.

rm responder action <name>

name

Name of the responder action to remove.

rm responder action act_before

Modifies the specified parameters of a responder action.

set responder action <name> [-target <string> [-bypassSafetyCheck (YES | NO)]] [-htmlpage <string>] [-comment <string>]

name

Name of the responder action to be modified.

target

Expression specifying what to respond with. Typically a URL for redirect policies or a default-syntax expression. In addition to NetScaler default-syntax expressions that refer to information in the request, a stringbuilder expression can contain text and HTML, and simple escape codes that define new lines and paragraphs. Enclose each stringbuilder expression element (either a NetScaler default-syntax expression or a string) in double quotation marks. Use the plus (+) character to join the elements.

Examples:

1) Respondwith expression that sends an HTTP 1.1 200 OK response:

"HTTP/1.1 200 OK\\r\\n\\r\\n"

2) Redirect expression that redirects user to the specified web host and appends the request URL to the redirect.

"http://backupsite2.com" + HTTP.REQ.URL

3) Respondwith expression that sends an HTTP 1.1 404 Not Found response with the request URL included in the response:

"HTTP/1.1 404 Not Found\\r\\n\\r\\n"+ "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."

The following requirement applies only to the NetScaler CLI:

Enclose the entire expression in single quotation marks. (NetScaler default expression elements should be included inside the single quotation marks for the entire expression, but do not need to be enclosed in double quotation marks.)

htmlpage

For respondwithhtmlpage policies, name of the HTML page object to use as the response. You must first import the page object.

comment

Comment. Any type of information about this responder action.

1. set responder action act_responder -target 'HTTP.REQ.HEADER(MYURL)' -bypassSafetyCheck YES/, 2. set responder action act_responder -htmlpage my-local-file

---

Use this command to remove responder action settings.Refer to the set responder action command for meanings of the arguments.

unset responder action <name> -comment

Displays the current settings for the specified responder action. If no action name is provided, displays a list of all responder actions currently configured on the NetScaler appliance, with abbreviated settings.

show responder action [<name>]

name

Name of the responder action.

summary

fullValues

format

level

stateflag

type

Type of responder action. It can be: (respondwith).

target

Expression specifying what to to respond with

htmlpage

Option specifying to respondwith htmlpage

bypassSafetyCheck

The safety check to allow unsafe expressions.

hits

The number of times the action has been taken.

referenceCount

The number of references to the action.

undefHits

The number of times the action resulted in UNDEF.

comment

Comment. Any type of information about this responder action.

builtin

Flag to determine whether responder action is built-in or not

devno

count

1. show responder action 2. show responder action act_insert

Renames a responder action.

rename responder action <name>@ <newName>@

name

Existing name of the responder action.

newName

New name for the responder action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder action" or my responder action').

rename responder action oldname newname

responder global

Sep 22, 2015

The following operations can be performed on "responder global":

bind | unbind | show

Activates the specified responder policy for all requests sent to the NetScaler appliance.

bind responder global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]

policyName

Name of the responder policy to activate. If you want to create the policy as well as activate it, specify a name for the responder policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy" or 'my responder policy').

i) bind responder global pol9 9

Unbind the specified responder policy from responder global.

unbind responder global <policyName> [-type <type>] [-priority <positive_integer>]

policyName

Name of the policy to unbind.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind responder global pol9

Displays the list of policies bound to the specified responder global bind point. If no bind point is specified, displays a list of all policies bound to responder global.

show responder global [-type <type>]

type

Specifies the bind point whose policies you want to display. Available settings function as follows:

* REQ_OVERRIDE - Request override. Binds the policy to the priority request queue.

* REQ_DEFAULT - Binds the policy to the default request queue.

* OTHERTCP_REQ_OVERRIDE - Binds the policy to the non-HTTP TCP priority request queue.

* OTHERTCP_REQ_DEFAULT - Binds the policy to the non-HTTP TCP default request queue..

* SIPUDP_REQ_OVERRIDE - Binds the policy to the SIP UDP priority response queue..

* SIPUDP_REQ_DEFAULT - Binds the policy to the SIP UDP default response queue.

* MSSQL_REQ_OVERRIDE - Binds the policy to the Microsoft SQL priority response queue..

* MSSQL_REQ_DEFAULT - Binds the policy to the Microsoft SQL default response queue.

* MYSQL_REQ_OVERRIDE - Binds the policy to the MySQL priority response queue.

* MYSQL_REQ_DEFAULT - Binds the policy to the MySQL default response queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT

summary

fullValues

format

level

stateflag

policyName

Name of the responder policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation, Available settings function as follows:

* vserver - Forward the request to the specified virtual server.

* policylabel - Invoke the specified policy label.

labelName

Name of the policy label to invoke. If the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is policylabel.

flowType

flowtype of the bound responder policy.

numpol

number of polices bound to label.

flags

devno

count

show responder global

responder htmlpage

Sep 22, 2015

The following operations can be performed on "responder htmlpage":

import | rm | update | show

Imports the specified HTML page to the NetScaler appliance, assigns it the specified name, and stores it in the list of Responder HTML page objects.

import responder htmlpage [<src>] <name> [-comment <string>] [-overwrite]

src

Local path to and name of, or URL \\(protocol, host, path, and file name\\) for, the file in which to store the imported HTML page.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the HTML page object on the NetScaler appliance.

comment

Any comments to preserve information about the HTML page object.

overwrite

Overwrites the existing file

import responder htmlpage http://www.example.com/page.html my-responder-page

---

Removes the specified HTML page object.

rm responder htmlpage <name>

name

Name of the HTML page object to remove.

rm responder htmlpage <name>

Updates the specified HTML page object from the source.

update responder htmlpage <name>

name

Name to assign to the HTML page object on the NetScaler appliance.

update responder htmlpage my-responder-page

Displays the specified HTML page object. If no HTML page object is specified, lists all HTML page objects on the NetScaler appliance.

show responder htmlpage [<name>]

name

Name of the HTML page object to display.

response

show responder htmlpage

---

responder param

Sep 22, 2015

The following operations can be performed on "responder param":

set | unset | show

Sets the default responder undefined action. If an UNDEF event is triggered during policy evaluation and if no undefAction is specified for the current policy, this value is used.

set responder param -undefAction <string>

undefAction

Action to perform when policy evaluation creates an UNDEF condition. Available settings function as follows:

* NOOP - Send the request to the protected server.

* RESET - Reset the request and notify the user's browser, so that the user can resend the request.

* DROP - Drop the request without sending a response to the user.

Default value: "NOOP"

set responder param -undefAction RESET

Resets the global undefAction to NOOP..Refer to the set responder param command for meanings of the arguments.

unset responder param -undefAction

unset responder param -undefAction

Displays the default responder undefAction.

show responder param

format

level

undefAction

Name of the responder action.

show responder param

responder policy

Sep 22, 2015

The following operations can be performed on "responder policy":

add | rm | set | unset | show | rename | stat

Creates a responder policy, which specifies requests that the NetScaler appliance intercepts and responds to directly instead of forwarding them to a protected server.

add responder policy <name> <rule> <action> [<undefAction>] [-comment <string>] [-logAction <string>] [-appflowAction <string>]

name

Name for the responder policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the responder policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy" or 'my responder policy').

rule

Default syntax expression that the policy uses to determine whether to respond to the specified request.

action

Name of the responder action to perform if the request matches this responder policy.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition.

Available settings function as follows:

* NOOP - Send the request to the protected server instead of responding to it.

* RESET - Reset the request and notify the user's browser, so that the user can resend the request.

* DROP - Drop the request without sending a response to the user.

comment

Any type of information about this responder policy.

logAction

Name of the messagelog action to use for requests that match this policy.

appflowAction

AppFlow action to invoke for requests that match this policy.

i) add responder policy pol9 "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh3\\\")" act_respondwith

---

Removes the specified responder policy.

rm responder policy <name>

name

Name of the responder policy to remove.

rm responder policy pol9

Modifies the rule or action portion of the specified responder policy.

set responder policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>] [-logAction <string>] [-appflowAction <string>]

name

Name of the responder policy.

rule

Default syntax expression that the policy uses to determine whether to respond to the specified request.

action

Name of the responder action to perform if the request matches this responder policy.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition.

Available settings function as follows:

* NOOP - Send the request to the protected server instead of responding to it.

* RESET - Reset the request and notify the user's browser, so that the user can resend the request.

* DROP - Drop the request without sending a response to the user.

comment

Any type of information about this responder policy.

logAction

Name of the messagelog action to use for requests that match this policy.

appflowAction

AppFlow action to invoke for requests that match this policy.

```
set responder policy pol9 -rule "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"
```

Removes the settings of an existing responder policy. Attributes for which a default value is available revert to their default values. See the set responder policy command for descriptions of the parameters..Refer to the set responder policy command for meanings of the arguments.

```
unset responder policy <name> [-undefAction] [-comment] [-logAction] [-appflowAction]
```

```
unset responder policy respol9 -undefAction
```

Displays the current settings for the specified responder policy. If no policy name is specified, displays a list of all responder policies currently configured on the NetScaler appliance, with abbreviated settings.

show responder policy [<name>] show responder policy stats - alias for 'stat responder policy'

name

Name of the responder policy for which to display settings.

summary

fullValues

format

level

stateflag

rule

Rule of the policy.

action

Responder action associated with the policy.

undefAction

UNDEF action associated with the policy.

hits

Number of hits.

undefHits

Number of policy UNDEF hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any type of information about this responder policy.

logAction

Name of the messagelog action to use for requests that match this policy.

bindPolicyType

vserverType

appflowAction

AppFlow action to invoke for requests that match this policy.

builtin

Flag to determine if responder policy is built-in or not

devno

count

show responder policy

Renames the specified responder policy.

rename responder policy <name>@ <newName>@

name

Existing name of the responder policy.

newName

New name for the responder policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.)

hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy" or 'my responder policy').

rename responder policy oldname newname

Displays statistics for all responder policies currently configured on the NetScaler appliance, or detailed statistics for the specified policy.

stat responder policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the responder policy for which to show detailed statistics.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

responder policylabel

Sep 22, 2015

The following operations can be performed on "responder policylabel":

add | rm | bind | unbind | show | stat | rename

Creates a user-defined responder policy label, to which you can bind policies. A policy label is a tool for evaluating a set of policies in a specified order. By using a policy label, you can configure the responder feature to choose the next policy, invoke a different policy label, or terminate policy evaluation completely by looking at whether the previous policy evaluated to TRUE or FALSE.

add responder policylabel <labelName> [-policylabeltype <policylabeltype>]

labelName

Name for the responder policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the responder policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy label" or my responder policy label').

policylabeltype

Type of responses sent by the policies bound to this policy label. Types are:

* HTTP - HTTP responses.

* OTHERTCP - NON-HTTP TCP responses.

* SIP_UDP - SIP responses.

* MYSQL - SQL responses in MySQL format.

* MSSQL - SQL responses in Microsoft SQL format.

Possible values: HTTP, OTHERTCP, SIP_UDP, MYSQL, MSSQL

Default value: NS_PLTMAP_RSP_REQ

```
add responder policylabel resp_lab
```

Removes a responder policy label.

rm responder policylabel <labelName>

labelName

Name of the responder policy label to remove.

rm responder policylabel resp_lab

Binds the specified responder policy to the specified policy label.

bind responder policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

labelName

Name of the responder policy label to which to bind the policy.

policyName

Name of the policy to bind to the responder policy label.

i) bind responder policylabel resp_lab pol_resp 1 2 ii) bind responder policylabel resp_lab pol_resp 1 2 -invoke vserver CURRENT

Unbinds the specified responder policy from the specified policy label.

unbind responder policylabel <labelName> <policyName> [-priority <positive_integer>]

labelName

Name for the responder policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the responder policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy label" or my responder policy label').

policyName

The name of the policy to be unbound.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind responder policylabel resp_lab pol_resp

Displays the current settings for the specified responder policy label. If no policy label is specified, displays a list of all responder policy labels currently configured on the NetScaler appliance, with abbreviated settings.

show responder policylabel [<labelName>]

labelName

Name of the responder policy label.

summary

fullValues

format

level

policylabeltype

The type of the policy label.

stateflag

numpol

number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the responder policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label and evaluate the specified policy label.

labelType

Type of policy label to invoke. Available settings function as follows:

* vserver - Invoke an unnamed policy label associated with a virtual server.

* policylabel - Invoke a user-defined policy label.

labelName

* If labelType is policylabel, name of the policy label to invoke.

* If labelType is reqvserver or resvserver, name of the virtual server.

flags

devno

count

i) show responder policylabel resp_lab ii) show responder policylabel

---

Displays statistics for the specified responder policy label. If no policy label name is provided, displays abbreviated statistics for all responder policy labels currently configured on the NetScaler appliance.

stat responder policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

labelName

Name of the responder policy label.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy Label Hits (Hits)

Number of times policy label was invoked.

Renames the specified responder policy label.

rename responder policylabel <labelName>@ <newName>@

labelName

Current name of the responder policy label.

newName

New name for the responder policy label. Must begin with a letter, number, or

the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

rename responder policylabel oldname newname

Router Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- router bgp
- router ospf
- router rip
- vtysh

router bgp

Sep 22, 2015

The following operations can be performed on "router bgp":

add | clear | rm | set | unset | show

NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

autonomousSystem

The BGP autonomous system.

Minimum value: 1

routerID

The router ID of the router.

learnRoute

The state of route learning from BGP.

staticRedistribute

The state of router in redistribution of static routes.

kernelRedistribute

The state of router in redistribution of kernel routes.

conRedistribute

The state of router in redistribution of connected routes.

neighbor

Add a BGP neighbor.

network

The neighbor to be advertised.

NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

neighbor

The neighbor associated with the connection that needs to be torn down.

all

Reset TCP connections to all neighbors.

NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

neighbor

To remove a particular neighbor.

NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

routerID

The Router ID of this router.

learnRoute

The state of the router in learning routes from BGP. Use this option to enable route learning and installation from BGP.

staticRedistribute

The state of the router in redistributing static routes. Use this option to enable the redistribution of static routes.

kernelRedistribute

The state of the router in redistribution of kernel routes.

conRedistribute

The state of the router in redistributing connected routes. Use this option to enable the redistribution of connected routes into the BGP domain.

neighbor

The IP address of a BGP peer for the router.

network

The network to be advertized.

Use this command to remove router bgp settings.Refer to the set router bgp command for meanings of the arguments.NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

---

NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

bgpOptions

option to show BGP command either neighbors or summary

Possible values: neighbors, summary

routeMap

The BGP route map.

summary

fullValues

format

level

devno

count

stateflag

router ospf

Sep 22, 2015

The following operations can be performed on "router ospf":

set | unset | show

Configure different OSPF parameters. NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

routerID

The router ID.

passiveInterface

The mode of the Interface. Use this option to change the mode of the interface to listen only.

staticRedistribute

The state of the router in redistributing static routes. Use this option to enable the redistribution of static routes.

kernelRedistribute

The state of the router in redistributing kernel routes. Use this option to enable the redistribution of kernel routes.

conRedistribute

The state of the router in redistributing connected routes. Use this option to enable the redistribution of connected routes.

learnRoute

The state of the router in learning routes from OSPF. Use this option to enable route learning from OSPF.

network

The broadcast network on which OSPF is to be run.

host

The stub link.

```
set ospf -routerID 1.2.3.4
```

Unset the OSPF parameters that were configured using the ###set ospf### command..Refer to the set router ospf command for meanings of the arguments.NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

unset ospf -router-id

Display the state of the OSPF daemon. NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

ospfoptions

The Router OSPF option. Use this option to display one of border-routers, database, interface, neighbor, route, and virtual-links.

Possible values: border-routers, database, interface, neighbor, route, virtual-links

format

level

network

The network on which OSPF is running.

netmask

Netmask of the network on which OSPF is running

show ospf neighbor

---

router rip

Sep 22, 2015

The following operations can be performed on "router rip":

set | unset | show

Configure the RIP daemon. NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

defaultMetric

The default metrics when advertising routes.

Default value: 1

Minimum value: 1

Maximum value: 16

passiveInterface

The mode of the interface to listen only.

learnRoute

The state of Route learning. Use this option to enable route learning and installation in the kernel.

staticRedistribute

The state of redistributing static routes.

kernelRedistribute

The state of redistributing kernel routes.

network

The broadcast network on which RIP must run.

set router rip -kernelRedistribute

Unset the RIP parameters..Refer to the set router rip command for meanings of the arguments.NOTE: This command is

deprecated.All routing configurations have now been moved to vtysh

unset rip -default-metric

Display the RIP configuration. NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

ripOptions

RIP option in show command, one of database or interface.

Possible values: database, interface

format

level

network

The broadcast network on which RIP must run.

netmask

show rip interface

---

vtysh

Sep 22, 2015

The following operations can be performed on "vtysh":

Enters into the Virtual Teletype Shell (VTYSH) prompt, at which you can configure all the dynamic routing protocols. The NetScaler dynamic routing suite is based on ZebOS, the commercial version of GNU Zebra.

vtysh

SureConnect Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- sc
- sc parameter
- sc policy
- sc stats

SC

Sep 22, 2015

The following operations can be performed on "sc":

Displays SureConnect statistics.

stat sc [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

SC condition triggered (ScTrigd)

Number of times that SureConnect conditions were triggered.

SC trigger condition failed

Total number of times SureConnect was not triggered because the thresholds conditions failed.

Policy matches

Total number of incoming requests that matched configured sureconnect policies.

SC responses sent

Total number of in-memory java script served which throws the pop-up window.

Reissued requests (ReissReq)

Total number of reissued SureConnect requests.

Valid reissued requests

Total number of requests that were handled in a single SureConnect session.

Alternate content requests

Total number of alternate content served which throws the pop-up window.

SC POST requests

Total number of HTTP POST requests that triggered SureConnect feature.

SC statistics timeout

Toal number of times that SureConnect statistics were reset.

Unsupported browsers

Total number of requests that came from all unsupported browsers.

Tampered SC cookies

Total number of corrupted SureConnect cookies.

sc parameter

Sep 22, 2015

The following operations can be performed on "sc parameter":

set | unset | show

Sets the parameters for displaying SureConnect information.

set sc parameter [-sessionLife <secs>] [-vsr <input_filename>]

sessionLife

Time, in seconds, between the first time and the next time the SureConnect alternative content window is displayed. The alternative content window is displayed only once during a session for the same browser accessing a configured URL, so this parameter determines the length of a session.

Default value: 300

Minimum value: 1

Maximum value: 4294967294

vsr

File containing the customized response to be displayed when the ACTION in the SureConnect policy is set to NS.

Default value: "DEFAULT"

set sc parameter -sessionlife 200 -vsr /etc/vsr.htm

Use this command to remove sc parameter settings.Refer to the set sc parameter command for meanings of the arguments.

unset sc parameter [-sessionLife] [-vsr]

Displays the values of the session life and vsr filename parameters.

show sc parameter

format

level

sessionLife

The time between first time the Sureconnect aternate content window displays and the next time it displays.The SureConnect alternate content window is displayed only once during a session. For the same browser accessing a configured URL.The value is in seconds.

vsr

The customized response will be displayed to the user if the alternate content server has been determined by the system to have failed.

If you have created a customized response that you want the system to use, enter its filename (if you renamed the vsr.htm file supplied by system). If you have not renamed the file, enter /etc/vsr.htm as the filename.

> show sc parameter Sure Connect Parameters: Sessionlife: 300 Vsr: DEFAULT Done

sc policy

Sep 22, 2015

The following operations can be performed on "sc policy":

add | rm | set | unset | show | stat

Creates a new SureConnect policy.

add sc policy <name> [-url <URL> | -rule <expression>] [-delay <usecs>] [-maxConn <positive_integer>] [-action <action> (<altContentSvcName> <altContentPath>)]

name

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

url

URL against which to match incoming client request.

rule

Expression against which the traffic is evaluated.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

delay

Delay threshold, in microseconds, for requests that match the policy's URL or rule. If the delay statistics gathered for the matching request exceed the specified delay, SureConnect is triggered for that request.

Minimum value: 1

Maximum value: 599999999

maxConn

Maximum number of concurrent connections that can be open for requests that match the policy's URL or rule.

Minimum value: 1

Maximum value: 4294967294

action

Action to be taken when the delay or maximum-connections threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service.

NS - Serve alternative content from the NetScaler appliance.

NO ACTION - Serve no alternative content. However, delay statistics are still collected for the configured URLs, and, if the Maximum Client Connections parameter is set, the number of connections is limited to the value specified by that parameter. (However, alternative content is not served even if the maxConn threshold is met).

Possible values: ACS, NS, NOACTION

altContentSvcName

Name of the alternative content service to be used in the ACS action.

altContentPath

Path to the alternative content service to be used in the ACS action.

add sc policy scpol_ns -delay 1000000 -url /delay.asp -action NS add policy expression exp_acs "url == /mc_acs.asp" add service svc_acs 10.110.100.253 http 80 add scp

---

Removes the specified SureConnect policy.

rm sc policy <name>

name

Name of the policy to be removed.

rm sc policy scpol_ns rm sc policy scpol_acs

Modifies the specified settings of a SureConnect policy.

set sc policy <name> [-url <URL> | -rule <expression>] [-delay <usecs>] [-maxConn <positive_integer>] [-action <action> (<altContentSvcName> <altContentPath>)]

name

Name of the policy to be modified.

url

URL against which to match requests. URLs take precedence over rules in SureConnect policies.

rule

Expression against which the traffic is evaluated.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

delay

Delay threshold, in microseconds, for requests that match the policy's URL or rule. If the delay statistics gathered for the matching request exceed the specified delay, SureConnect is triggered for that request.

Minimum value: 1

Maximum value: 599999999

maxConn

Maximum number of concurrent connections that can be open for the configured URL or rule.

Minimum value: 1

Maximum value: 4294967294

action

Action to be taken when the delay or maximum-connections threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service.

NS - Serve alternative content from the NetScaler appliance.

NO ACTION - Serve no alternative content. However, delay statistics are still collected for the configured URLs, and, if the Maximum Client Connections parameter is set, the number of connections is limited to the value specified by that parameter. (However, alternative content is not served even if the maxConn threshold is met).

Possible values: ACS, NS, NOACTION

set sc policy scpol_ns -delay 2000000 set sc policy scpol_acs -maxconn 100

Use this command to remove sc policy settings.Refer to the set sc policy command for meanings of the arguments.

unset sc policy <name> [-delay] [-maxConn]

Displays information about the SureConnect policies.

show sc policy [<name>]

name

Name of a policy about which to display detailed information. To display information about all the SureConnect policies, do not set this parameter.

summary

fullValues

format

level

url

The URL name. The system matches the incoming client request against the URL you enter here.

rule

The rule that the system matches with the incoming request.

The system matches the incoming request against the rules you enter here. Before matching against the configured rules, the NetScaler 9000 system matches the requests with any of the configured URLs. Thus, URLs have a higher precedence over rules. If the incoming request does not match any of the configured URLs or the rules that have been configured, then SureConnect does not trigger.

Expression logic is expression names, separated by the logical operators | | and && , and possibly grouped using parenthesis. If the expression contains blanks (for example, between an expression name and a logical operator), then the entire argument must be enclosed in double quotes.The following are valid expression logic:

ns_ext_cgi| |ns_ext_asp

ns_non_get && (ns_header_cookie| |ns_header_pragma)

delay

Delay threshold, in microseconds, for requests that match the policy's URL or rule. If the delay statistics gathered for the matching request exceed the specified delay, SureConnect is triggered for that request.

maxConn

Maximum number of concurrent connections that can be open for requests that match the policy's URL or rule.

action

The action to be taken when the thresholds are met. The valid options are ACS , NS and NOACTION .

ACS - Specifies that alternate content is to be served from altContSvcName with the path altContPath .

NS - Specifies that alternate content is to be served from the NetScaler 9000 system. See the set sc parameter command to customize the response served from the system.

NOACTION - Specifies that no alternate content is to be served. However, delay statistics are still collected for the configured URLs. If the - maxconn argument is specified, the number of connections is limited to that specified value for that configured URL or rule (alternate content will not served even if the - maxconn threshold is met).

altContentSvcName

Name of the alternative content service to be used in the ACS action.

altContentPath

Path to the alternative content service to be used in the ACS action.

devno

count

stateflag

> show sc policy 2 monitored Sure Connect Policies: 1) Name: scpol_ns RULE: exp1 Delay: 1000000 microsecs A

Displays statistics about SureConnect policies.

stat sc policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the policy about which to display statistics. To display statistics about all SureConnect policies, do not set this parameter.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Server TTLB (SvrTTLB)

Server Time-To-Last-Byte in seconds calculated for this SureConnect policy.

Average server TTLB

Average server transaction time in seconds for this SureConnect Policy.

Average client TTLB (AvClTTLB)

Average value of the client Time-To-Last-Byte in seconds for this SureConnect policy.

Physical service IP (SvcIP)

IP address of the service in dotted notation for which these statistics are maintained.

Physical service port (SvcPort)

Port of the service for which these statistics are maintained.

Current client connections (CurClts)

Number of clients currently allowed a server connection by this SureConnect policy.

Current SC queue length (WaitClts)

Current number of SureConnect priority clients that are waiting for a server connection.

Current server connections (CurSvrs)

Current number of open connections to the servers matching this policy.

Estimated waiting time (Sec) (WaitTime)

Value of the currently estimated waiting time in seconds for the configured URL.

Client TCP connections (TotClt)

Total number of clients that were allowed a server connection by this SureConnect policy.

Server TCP connections (TotSvr)

Total number of server connections that were established through this SureConnect policy.

Client HTTP transactions

Total number of client transactions processed by this SureConnect policy.

Server HTTP transactions (SrvTrans)

Number of 200 OK responses received from the web server by this SureConnect policy.

Requests received (TotReq)

Total number of requests received by this SureConnect policy.

Request bytes received (ReqBytes)

Total number of request bytes received by this SureConnect policy.

Server responses received (TotResp)

Total number of server responses received by this SureConnect policy.

Response bytes received (RspBytes)

Total number of response bytes received by this SureConnect policy.

sc stats

Sep 22, 2015

The following operations can be performed on "sc stats":

show sc stats is an alias for stat sc

show sc stats - alias for 'stat sc'

SNMP Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- snmp
- snmp alarm
- snmp community
- snmp engineId
- snmp group
- snmp manager
- snmp mib
- snmp oid
- snmp option
- snmp stats
- snmp trap
- snmp user
- snmp view

snmp

Sep 22, 2015

The following operations can be performed on "snmp":

Display the statistics related to SNMP.

stat snmp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

SNMP packets received (PktsRx)

SNMP packets received.

SNMP packets sent (PktsTx)

SNMP packets transmitted.

Get requests receieved (GetReqRx)

SNMP Get-Request PDUs that have been accepted and processed.

Get-next requests receieved (GtNextRx)

SNMP Get-Next PDUs that have been accepted and processed.

Get-bulk requests received (GtBulkRx)

SNMP Get-Bulk PDUs that have been accepted and pro?cessed.

Responses sent (RspTx)

SNMP Get-Response PDUs that have been generated by the NetScaler.

Traps messages sent (TrapsTx)

SNMP Trap PDUs that have been generated by the NetScaler.

Requests dropped (ReqDrop)

SNMP requests dropped.

ASN.1/BER errors in requests (PrsErrRx)

Number of ASN.1 or BER errors encountered when decoding received SNMP Messages.

Unsupported SNMP version (UnkVrsRx)

Number of SNMP messages received, which were for an unsupported SNMP version.

Unknown community name (UnkCNRx)

SNMP messages received, which used an SNMP community name not known to the NetScaler.

No permission on community (BadCURx)

The total number of SNMP Messages received that represented an SNMP operation which was not allowed by the SNMP community named in the Message.

Unsupported security level (UnkSecLv)

SNMP packets that were dropped because they requested a security level that was

unknown to the NetScaler or otherwise unavailable.

Not in time window (NtTimeWd)

SNMP packets that were dropped because they appeared outside of the authoritative SNMP engine's window.

Unknown user name (UnkUser)

SNMP packets that were dropped because they referenced a user that was not known to the SNMP engine.

Unknown engine Id (UnkEngId)

SNMP packets that were dropped because they referenced an SNMP engine ID that was not known to the NetScaler.

Wrong digest value (WrgDgst)

SNMP packets that were dropped because they did not contain the expected digest value.

Decryption errors (DcrptErr)

SNMP packets that were dropped because they could not be decrypted.

stat snmp

snmp alarm

Sep 22, 2015

The following operations can be performed on "snmp alarm":

set | unset | enable | disable | show

Configures an SNMP alarm. You must enable and configure alarms to generate enterprise-specific trap messages. The NetScaler appliance sends these trap messages only to trap listeners of type (class) SPECIFIC. The SNMP alarms are either event based or threshold based. The NetScaler appliance supports the following user configurable alarms: HA-STATE-CHANGE: Change to primary/secondary CPU-USAGE: Individual CPU usage AVERAGE-CPU: Average CPU usage MGMT-CPU: Management CPU usage ENTITY-STATE: Entity state change SYNFLOOD: Global unacknowledged SYN count MEMORY: Memory usage VSERVER-REQRATE: Vserver specific request rate SERVICE-REQRATE: Service specific request rate ENTITY-RXRATE: Entity specific Rx bytes per sec ENTITY-TXRATE: Entity specific Tx bytes per sec ENTITY-SYNFLOOD: Entity specific unacknowledged SYN count CONFIG-CHANGE: System configuration changed SERVICE-MAXCLIENTS: Service hit max-client limit CONFIG-SAVE: System configuration was saved SERVICEGROUP-MEMBER-REQRATE: Request rate on a service group member SERVICEGROUP-MEMBER-MAXCLIENTS: Service group member hits max-client MONITOR-RTO-THRESHOLD: Monitor probe response timeout LOGIN-FAILURE: GUI/CLI/API login failure SSL-CERT-EXPIRY: Certificate expiry FAN-SPEED-LOW: Low fan speed VOLTAGE-LOW: Low voltage VOLTAGE-HIGH: High Voltage TEMPERATURE-HIGH: High temperature CPU-TEMPERATURE-HIGH: High CPU temperature POWER-SUPPLY-FAILURE: Power supply failure DISK-USAGE-HIGH: High disk usage INTERFACE-THROUGHPUT-LOW: Low Interface throughput MON_PROBE_FAILED: Monitor probe failure HA-VERSION-MISMATCH: HA netscaler's OS version mismatch HA-SYNC-FAILURE: HA config synchronization failure HA-NO-HEARTBEATS: No HA hearbeats HA-BAD-SECONDARY-STATE: Secondary state DOWN/UNKNOWN/STAY SECONDARY INTERFACE-BW-USAGE: System aggregate BW usage RATE-LIMIT-THRESHOLD-EXCEEDED: Client exceed rate-limit threshold ENTITY-NAME-CHANGE: Entity name change HA-PROP-FAILURE: HA config propagation failure IP-CONFLICT: IP conflict PF-RL-RATE-THRESHOLD: Platform rate limit in Mbps PF-RL-PPS-THRESHOLD: Platform packets per second limit PF-RL-RATE-PKTS-DROPPED: Packet Drops due to platform rate limit PF-RL-PPS-PKTS-DROPPED: Packet Drops due to platform packet per sec limit APPFW-START-URL: AppFirewall Start URL violation APPFW-DENY-URL: AppFirewall Deny URL violation APPFW-REFERER-HEADER: AppFirewall Referer Header violation APPFW-CSRF-TAG: AppFirewall CSRF Tag violation APPFW-COOKIE: AppFirewall Cookie violation APPFW-FIELD-CONSISTENCY: AppFirewall Field Consistency violation APPFW-BUFFER-OVERFLOW: AppFirewall Buffer Overflow violation APPFW-FIELD-FORMAT: AppFirewall Field Format violation APPFW-SAFE-COMMERCE: AppFirewall Safe Commerce violation APPFW-SAFE-OBJECT: AppFirewall Safe Object violation APPFW-POLICY-HIT: AppFirewall Policy Hit APPFW-VIOLATIONS-TYPE: AppFirewall Content Type violation APPFW-XSS: AppFirewall Cross Site Scripting violation APPFW-XML-XSS: AppFirewall XML Cross Site Scripting violation APPFW-SQL: AppFirewall SQL violation APPFW-XML-SQL: AppFirewall XML SQL violation APPFW-XML-ATTACHMENT: AppFirewall XML Attachment violation APPFW-XML-DOS: AppFirewall XML DoS violation APPFW-XML-VALIDATION: AppFirewall XML Validation violation APPFW-XML-WSI: AppFirewall XML WSI violation APPFW-XML-SCHEMA-COMPILE: AppFirewall XML Schema Compile violation APPFW-XML-SOAP-FAULT: AppFirewall XML Soap Fault violation DNSKEY-EXPIRY: DNSKEY expiry DATASTREAM-RATE-LIMIT-HIT: DataStream Rate Limit Hit HA-LICENSE-MISMATCH: HA netscaler's license mismatch SSL-CARD-FAILED: SSL Card Failed SSL-CARD-NORMAL: SSL Card Normal WARM-RESTART-EVENT: Warm Restart Event Occurred HARD-DISK-DRIVE-ERRORS: Hard Disk Drive Errors COMPACT-FLASH-ERRORS: Compact Flash Errors CALLHOME-UPLOAD-EVENT: Attempt to upload Show Tech Support Archive 1024KEY-EXCHANGE-RATE: 1024 Key Exchange Rate 2048KEY-EXCHANGE-RATE: 2048 Key Exchange Rate 4096KEY-EXCHANGE-RATE: 4096 Key Exchange Rate SSL-CUR-SESSION-INUSE: SSL Current Sessions In Use CLUSTER-NODE-

HEALTH: Cluster Node Health State Change CLUSTER-NODE-QUORUM: Cluster Node View has Quorum CLUSTER-VERSION-MISMATCH: Cluster Node Version Mismatch CLUSTER-CCO-CHANGE: Cluster Configuration Coordinator Change CLUSTER-OVS-CHANGE: Cluster Operational View Set Change CLUSTER-SYNC-FAILURE: Cluster Config Synchronization Failure CLUSTER-PROP-FAILURE: Cluster Config Propagation Failure HA-STICKY-PRIMARY: Fixed primary state owing to max HA flips INBAND-PROTOCOL-VERSION-MISMATCH: Inband protocol mismatch between BR and QoSd SSL-CHIP-REINIT: SSL Chip Reinit VRID-STATE-CHANGE: VRID State Change For the purposes of this command, entity includes vservers and services.

set snmp alarm <trapName> [-thresholdValue <positive_integer> [-normalValue <positive_integer>]] [-time <secs>] [-state (ENABLED | DISABLED)] [-severity <severity>] [-logging (ENABLED | DISABLED)]

trapName

Name of the SNMP alarm. This parameter is required for identifying the SNMP alarm and cannot be modified.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, DATASTREAM-RATE-LIMIT-HIT, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE

thresholdValue

Value for the high threshold. The NetScaler appliance generates an SNMP trap message when the value of the attribute associated with the alarm is greater than or equal to the specified high threshold value.

Minimum value: 1

time

Interval, in seconds, at which the NetScaler appliance generates SNMP trap messages when the conditions

specified in the SNMP alarm are met.Can be specified for the following alarms: SYNFLOOD, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS,HA-BAD-SECONDARY-STATE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, PORT-ALLOC-FAILED and APPFW traps. Default trap time intervals: SYNFLOOD and APPFW traps = 1sec, PORT-ALLOC-FAILED = 3600sec(1 hour), Other Traps = 86400sec(1 day)

Default value: 1

state

Current state of the SNMP alarm. The NetScaler appliance generates trap messages only for SNMP alarms that are enabled. Some alarms are enabled by default, but you can disable them.

Possible values: ENABLED, DISABLED

Default value: ENABLED

severity

Severity level assigned to trap messages generated by this alarm. The severity levels are, in increasing order of severity, Informational, Warning, Minor, Major, and Critical.

This parameter is useful when you want the NetScaler appliance to send trap messages to a trap listener on the basis of severity level. Trap messages with a severity level lower than the specified level (in the trap listener entry) are not sent.

Possible values: Critical, Major, Minor, Warning, Informational

Default value: SNMP_SEV_UNKNOWN

logging

Logging status of the alarm. When logging is enabled, the NetScaler appliance logs every trap message that is generated for this alarm.

Possible values: ENABLED, DISABLED

Default value: ENABLED

```
set snmp alarm VSERVER-REQRATE -thresholdValue 10000 -normalValue 100
```

Resets the specified parameters of an SNMP alarm to their default settings..Refer to the set snmp alarm command for meanings of the arguments.

```
unset snmp alarm <trapName> [-thresholdValue] [-normalValue] [-time] [-state] [-severity] [-logging]
```

unset snmp alarm VSERVER-REQRATE

Enables or disables an SNMP alarm. The NetScaler appliance looks for conditions specified in the enabled SNMP alarms. When the condition in any enabled SNMP alarm is met, the appliance generates an SNMP trap message. It does not look for conditions specified in disabled SNMP alarms and therefore does not generate an SNMP trap message when the condition in any disabled SNMP alarm is met. Some alarms are enabled by default, but you can disable them.

enable snmp alarm <trapName> ...

trapName

Name of the SNMP alarm. This parameter is required for identifying the SNMP alarm.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, DATASTREAM-RATE-LIMIT-HIT, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE

enable snmp alarm VSERVER-REQRATE enable snmp alarm CPU SYNFLOOD

Disables an SNMP alarm. The NetScaler appliance does not generate trap messages for SNMP alarms that are disabled. Some alarms are enabled by default, but you can disable them.

disable snmp alarm <trapName> ...

trapName

Name of the SNMP alarm. This parameter is required for identifying the SNMP alarm.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, DATASTREAM-RATE-LIMIT-HIT, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE

disable snmp alarm VSERVER-REQRATE disable snmp alarm CPU SYNFLOOD

Displays the settings of all SNMP alarms or of the specified SNMP alarm. To display the settings of all the SNMP alarms, run the command without any parameters. To display the settings of a particular SNMP alarm, specify the trapName (Alarm name) of the SNMP alarm.

show snmp alarm [<trapName>]

trapName

Name of the SNMP alarm whose details you want the NetScaler appliance to display.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, DATASTREAM-RATE-LIMIT-HIT, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE

summary

fullValues

format

level

thresholdValue

The high threshold value.

normalValue

The normal threshold value.

time

The time interval for the SYNFLOOD alarm.

state

Current state of the SNMP alarm. The NetScaler appliance generates trap messages only for SNMP alarms that are enabled. Some alarms are enabled by default, but you can disable them.

severity

The severity of this alarm.

logging

The log status of the alarm.

flags

timeout

If DB is enabled and clear config is fired, then to reset timeinterval of alarm, corresponding default time value is needed. This hidden argument holds the default time value for the corresponding alarm.

devno

count

stateflag

snmp community

Sep 22, 2015

The following operations can be performed on "snmp community":

add | rm | show

Creates an SNMP community, which is a password (string) used to authenticate SNMP queries from SNMP managers. You can associate it with any of the following SNMP query types: GET, GET NEXT, ALL, GET BULK. You can associate one or more community strings with each query type. For example, if you associate two community strings, such as Example and Test, with the query type GET NEXT, the NetScaler appliance considers only those GET NEXT SNMP query packets that contain Example or Test as the community string.

add snmp community <communityName> <permissions>

communityName

The SNMP community string. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and special characters.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

permissions

The SNMP V1 or V2 query-type privilege that you want to associate with this SNMP community.

Possible values: GET, GET_NEXT, GET_BULK, SET, ALL

add snmp community public ALL add snmp community a#12ab GET_BULK

Removes an SNMP community from the NetScaler appliance. After you remove the SNMP community, the appliance does not respond to any SNMP queries that contain that community string.

rm snmp community <communityName>

communityName

The name of the SNMP community.

rm snmp community public

Displays the SNMP v1 or v2 query-type privileges (such as GET, GET NEXT, ALL, or GET BULK) that have been set for all SNMP communities or for the specified SNMP community. To display the settings of all the SNMP communities, run the command without any parameters. To display the settings of a particular SNMP community, specify the name of the SNMP community.

show snmp community [<communityName>]

communityName

The name of the SNMP community whose SNMP v1 or v2 query type privilege setting, such as GET, GET NEXT, ALL, or GET BULK, you want the NetScaler appliance to display.

summary

fullValues

format

level

permissions

The SNMP V1 or V2 query-type privilege that you want to associate with this SNMP community.

devno

count

stateflag

show snmp community

---

snmp engineId

Sep 22, 2015

The following operations can be performed on "snmp engineId":

set | unset | show

set snmp engineId

Modifies the SNMPv3 engine identification (ID) on the NetScaler appliance. Caution: Changing the ID of the SNMPv3 engine invalidates the current SNMP users. You have to reconfigure the SNMP users in the SNMP managers. The SNMPv3 engine has an identification (ID) that uniquely identifies it on the appliance and is used in the communication between the SNMPv3 user and the SNMPv3 engine. The engine ID is preconfigured by Citrix and is based on the MAC address of one of its interfaces. Overriding the engine ID is not necessary, but you can change it.

Synopsys

set snmp engineId <engineID> [-ownerNode <positive_integer>]

Arguments

engineID

A hexadecimal value of at least 10 characters, uniquely identifying the engineid

ownerNode

ID of the cluster node for which you are setting the engineid

Default value: -1

Maximum value: 31

---

unset snmp engineId

Resets the SNMPv3 engine identification (ID) on the NetScaler appliance to its default value. The NetScaler appliance derives the engine ID from the MAC address of one of its interfaces. Caution: Changing the ID of the SNMPv3 engine invalidates the current SNMP users. You have to reconfigure the SNMP users in the SNMP managers..Refer to the set snmp engineId command for meanings of the arguments.

Synopsys

unset snmp engineId [-ownerNode <positive_integer>]

---

show snmp engineId

Displays the ID of the SNMPv3 engine of the NetScaler appliance.

Synopsys

show snmp engineId [-ownerNode <positive_integer>]

Arguments

ownerNode

ID of the cluster node for which you are setting the engineid

Default value: -1

Maximum value: 31

format

level

Outputs

engineID

A hexadecimal value of at least 10 characters, uniquely identifying the engineid

defaultEngineID

Unique identifier to assign to the SNMPv3 engine. Should be a hexadecimal value with a minimum length of 10 hex characters.

devno

count

stateflag

snmp group

Sep 22, 2015

The following operations can be performed on "snmp group":

add | rm | set | show

add snmp group

Adds an SNMPv3 user group on the NetScaler appliance. SNMPv3 groups are logical aggregations of SNMPv3 users. SNMPv3 groups are used to implement access control and define the security levels for the users. You can add a maximum of 1000 SNMPv3 groups to the NetScaler appliance.

Synopsys

add snmp group <name> <securityLevel> -readViewName <string>

Arguments

name

Name for the SNMPv3 group. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the SNMPv3 group.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my name" or 'my name').

securityLevel

Security level required for communication between the NetScaler appliance and the SNMPv3 users who belong to the group. Specify one of the following options:

noAuthNoPriv. Require neither authentication nor encryption.

authNoPriv. Require authentication but no encryption.

authPriv. Require authentication and encryption.

Note: If you specify authentication, you must specify an encryption algorithm when you assign an SNMPv3 user to the group. If you also specify encryption, you must assign both an authentication and an encryption algorithm for each group member.

Possible values: noAuthNoPriv, authNoPriv, authPriv

readViewName

Name of the configured SNMPv3 view that you want to bind to this SNMPv3 group. An SNMPv3 user bound to this group can access the subtrees that are bound to this SNMPv3 view as type INCLUDED, but cannot access the ones that are type EXCLUDED. If the NetScaler appliance has multiple SNMPv3 view entries with the same name,

all such entries are associated with the SNMPv3 group.

---

rm snmp group

Removes an SNMPv3 group entry from the NetScaler appliance. The appliance can have multiple SNMPv3 groups with the same name, differentiated by the securityLevel (Security level) parameter setting. Therefore, to identify an SNMPv3 group entry that you want to remove, you have to specify both the name and security level of the SNMPv3 group.

Synopsys

rm snmp group <name> <securityLevel>

Arguments

name

Name of the SNMPv3 group.

securityLevel

Security level of the SNMPv3 group.

Possible values: noAuthNoPriv, authNoPriv, authPriv

---

set snmp group

Modifies the specified parameters of an SNMPv3 group entry on the NetScaler appliance.

Synopsys

set snmp group <name> <securityLevel> -readViewName <string>

Arguments

name

The name specified in the SNMPv3 group entry that you want to modify. This parameter cannot be modified.

securityLevel

Security level required for communication between the NetScaler appliance and the SNMPv3 users who belong to the group. Specify one of the following options:

noAuthNoPriv. Require neither authentication nor encryption.

authNoPriv. Require authentication but no encryption.

authPriv. Require authentication and encryption.

Note: If you specify authentication, you must specify an encryption algorithm when you assign an SNMPv3 user to the group. If you also specify encryption, you must assign both an authentication and an encryption algorithm for

each group member.

Possible values: noAuthNoPriv, authNoPriv, authPriv

readViewName

Name of the configured SNMPv3 view that you want to bind to this SNMPv3 group. An SNMPv3 user bound to this group can access the subtrees that are bound to this SNMPv3 view as type INCLUDED, but cannot access the ones that are type EXCLUDED. If the NetScaler appliance has multiple SNMPv3 view entries with the same name, all such entries are associated with the SNMPv3 group.

---

show snmp group

Displays the settings of all SNMPv3 groups or of the specified SNMPv3 group. To display the settings of all SNMPv3 groups, run the command without any parameters. To display the settings of a particular SNMPv3 group, specify the name of the SNMPv3 group and securityLevel (Security level). The NetScaler appliance can have multiple SNMPv3 groups with the same name, differentiated by the securityLevel (Security level) parameter setting.

Synopsys

show snmp group [<name> <securityLevel>]

Arguments

name

Name of the SNMPv3 group whose details you want the NetScaler appliance to display.

securityLevel

Security level of the SNMPv3 group whose details you want the NetScaler appliance to display.

Possible values: noAuthNoPriv, authNoPriv, authPriv

summary

fullValues

format

level

Outputs

readViewName

Name of the configured SNMPv3 view that you want to bind to this SNMPv3 group. An SNMPv3 user bound to this group can access the subtrees that are bound to this SNMPv3 view as type INCLUDED, but cannot access the ones that are type EXCLUDED. If the NetScaler appliance has multiple SNMPv3 view entries with the same name, all such entries are associated with the SNMPv3 group.

storageType

The storage type for this group.

status

The status of this group.

devno

count

stateflag

snmp manager

Sep 22, 2015

The following operations can be performed on "snmp manager":

add | rm | set | unset | show

add snmp manager

Specifies an SNMP manager to query the NetScaler appliance. The added manager complies with SNMP V1, V2, and V3. If you specify one or more SNMP managers, the appliance does not accept SNMP queries from any hosts except the specified SNMP managers. You can specify up to a maximum of 100 IP based SNMP managers or networks and a maximum of 5 host-name based SNMP managers.

Synopsys

add snmp manager <IPAddress> ... [-netmask <netmask>] [-domainResolveRetry <integer>]

Arguments

IPAddress

IP address of the SNMP manager. Can be an IPv4 or IPv6 address. You can instead specify an IPv4 network address or IPv6 network prefix if you want the NetScaler appliance to respond to SNMP queries from any device on the specified network. Alternatively, instead of an IPv4 address, you can specify a host name that has been assigned to an SNMP manager. If you do so, you must add a DNS name server that resolves the host name of the SNMP manager to its IP address.

Note: The NetScaler appliance does not support host names for SNMP managers that have IPv6 addresses.

netmask

Subnet mask associated with an IPv4 network address. If the IP address specifies the address or host name of a specific host, accept the default value of 255.255.255.255.

Default value: 0xFFFFFFFF

domainResolveRetry

Amount of time, in seconds, for which the NetScaler appliance waits before sending another DNS query to resolve the host name of the SNMP manager if the last query failed. This parameter is valid for host-name based SNMP managers only. After a query succeeds, the TTL determines the wait time.

Minimum value: 5

Maximum value: 20939

Example

```
add snmp manager 192.168.1.20 192.168.2.42 add snmp manager 192.168.2.16 -netmask 255.255.255.240 add snmp manager hostnamemanager.com
```

---

rm snmp manager

Removes an SNMP manager from the list of managers that are allowed to access the NetScaler appliance.

Synopsys

rm snmp manager <IPAddress> ... [-netmask <netmask>]

Arguments

IPAddress

IPv4 or IPv6 address (or IPv4 host name) of the SNMP manager, or the IPv4 network address or IPv6 network prefix of the SNMP managers.

netmask

Subnet mask associated with an IPv4 SNMP manager entry. For a specific host, the subnet mask is 255.255.255.255.

Default value: 0xFFFFFFFF

Example

```
rm snmp manager 192.168.1.20 rm snmp manager 192.168.2.16 -netmask 255.255.255.240 rm snmp manager hostnamemanager.com
```

---

set snmp manager

Modifies the Domain Resolve Retry parameter of any host-name based SNMP manager configured on the NetScaler appliance.

Synopsys

set snmp manager <IPAddress> [-netmask <netmask>] [-domainResolveRetry <integer>]

Arguments

IPAddress

Host name of the SNMP manager for which you want to modify the Domain Resolve Retry parameter.

netmask

Subnet mask associated with an IPv4 network address. If the IP address specifies the address or host name of a specific host, accept the default value of 255.255.255.255.

Default value: 0xFFFFFFFF

domainResolveRetry

Amount of time, in seconds, for which the NetScaler appliance waits before sending another DNS query to resolve the host name of the SNMP manager if the last query failed. This parameter is valid for host-name based SNMP managers only. After a query succeeds, the TTL determines the wait time.

Minimum value: 5

Maximum value: 20939

Example

```
set snmp manager www.example.com -domainResolveRetry 7
```

---

unset snmp manager

Use this command to remove snmp manager settings.Refer to the set snmp manager command for meanings of the arguments.

Synopsys

unset snmp manager <IPAddress> -netmask <netmask> -domainResolveRetry

---

show snmp manager

Displays configuration information about all SNMP managers on the NetScaler appliance, or detailed information about the specified manager.

Synopsys

show snmp manager [<IPAddress> [-netmask <netmask>]]

Arguments

IPAddress

IPv4 or IPv6 address (or IPv4 host name) of the SNMP manager, or the IPv4 network address or IPv6 network prefix of the SNMP managers, about which to display information.

summary

fullValues

format

level

Outputs

IP

The resolved IP address of the hostname manager

domain

IP address of manager. It will be zero for hostname manager

domainResolveRetry

Amount of time, in seconds, for which the NetScaler appliance waits before sending another DNS query to resolve the host name of the SNMP manager if the last query failed. This parameter is valid for host-name based SNMP managers only. After a query succeeds, the TTL determines the wait time.

devno

count

stateflag

Example

show snmp manager

snmp mib

Sep 22, 2015

The following operations can be performed on "snmp mib":

set | unset | show

set snmp mib

Configures the SNMP agent of the NetScaler appliance with information that identifies the appliance, such as the name of the administrator for this NetScaler appliance, a name for the appliance, and the location of the appliance. SNMP managers can query the NetScaler appliance for this information.

Synopsys

set snmp mib [-contact <string>] [-name <string>] [-location <string>] [-customID <string>]

Arguments

contact

Name of the administrator for this NetScaler appliance. Along with the name, you can include information on how to contact this person, such as a phone number or an email address. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the information includes one or more spaces, enclose it in double or single quotation marks (for example, "my contact" or 'my contact').

Default value: "WebMaster (default)"

name

Name for this NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my name" or 'my name').

Default value: "NetScaler"

location

Physical location of the NetScaler appliance. For example, you can specify building name, lab number, and rack number. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the location includes one or more spaces, enclose it in double or single quotation marks (for example, "my location" or 'my location').

Default value: "POP (default)"

customID

Custom identification number for the NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a custom identification that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the ID includes one or more spaces, enclose it in double or single quotation marks (for example, "my ID" or 'my ID').

Default value: "Default"

---

unset snmp mib

Use this command to remove snmp mib settings.Refer to the set snmp mib command for meanings of the arguments.

Synopsys

unset snmp mib [-contact] [-name] [-location] [-customID]

---

show snmp mib

Displays the information that has been configured on the SNMP agent for the purpose of identifying the NetScaler appliance, such as the name of the appliance, administrator, and location.

Synopsys

show snmp mib

Arguments

format

level

Outputs

contact

Name of the administrator for this NetScaler appliance. Along with the name, you can include information on how to contact this person, such as a phone number or an email address. Can consist of 1 to 127

characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the information includes one or more spaces, enclose it in double or single quotation marks (for example, "my contact" or 'my contact').

name

Name for this NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my name" or 'my name').

location

Physical location of the NetScaler appliance. For example, you can specify building name, lab number, and rack number. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the location includes one or more spaces, enclose it in double or single quotation marks (for example, "my location" or 'my location').

sysDesc

The description of the system.

sysUptime

The UP time of the system in 100th of a second.

sysServices

The services offered by the system.

sysOID

The OID of the system's management system.

customID

Custom identification number for the NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a custom identification that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the ID includes one or more spaces, enclose it in double or single quotation marks (for example, "my ID" or 'my ID').

Example

```
show snmp mib
```

snmp oid

Sep 22, 2015

The following operations can be performed on "snmp oid":

show snmp oid

Displays the corresponding SNMP OIDs for the virtual servers, services, and service groups configured on the NetScaler appliance. To display the SNMP OID of all entities of a particular type, such as virtual servers, run the command with only that entity type specified. To display the SNMP of a particular entity, specify the entity type and the entity name.

Synopsys

show snmp oid <entityType> [<name>]

Arguments

entityType

The type of entity whose SNMP OIDs you want to displayType of entity whose SNMP OIDs you want the NetScaler appliance to display.

Possible values: VSERVER, SERVICE, SERVICEGROUP

name

Name of the entity whose SNMP OID you want the NetScaler appliance to display.

summary

fullValues

Outputs

snmpOID

The snmp oid.

stateflag

state flag

devno

count

Example

show snmp oid VSERVER vs1

snmp option

Sep 22, 2015

The following operations can be performed on "snmp option":

set | unset | show

set snmp option

Enables or disables SNMP options for SNMP SET and SNMP trap logging.

Synopsys

set snmp option [-snmpset (ENABLED | DISABLED)] [-snmpTrapLogging (ENABLED | DISABLED)]

Arguments

snmpset

Accept SNMP SET requests sent to the NetScaler appliance, and allow SNMP managers to write values to MIB objects that are configured for write access.

Possible values: ENABLED, DISABLED

Default value: DISABLED

snmpTrapLogging

Log any SNMP trap events (for SNMP alarms in which logging is enabled) even if no trap listeners are configured. With the default setting, SNMP trap events are logged if at least one trap listener is configured on the appliance.

Possible values: ENABLED, DISABLED

Default value: DISABLED

---

unset snmp option

Use this command to remove snmp option settings.Refer to the set snmp option command for meanings of the arguments.

Synopsys

unset snmp option [-snmpset] [-snmpTrapLogging]

---

show snmp option

Displays the settings for the following SNMP options: SNMP SET and SNMP trap Logging.

Synopsys

show snmp option

Arguments

format

level

Outputs

snmpset

Accept SNMP SET requests sent to the NetScaler appliance, and allow SNMP managers to write values to MIB objects that are configured for write access.

snmpTrapLogging

Log any SNMP trap events (for SNMP alarms in which logging is enabled) even if no trap listeners are configured. With the default setting, SNMP trap events are logged if at least one trap listener is configured on the appliance.

snmp stats

Sep 22, 2015

The following operations can be performed on "snmp stats":

show snmp stats

show snmp stats is an alias for stat snmp Displays the statistics related to SNMP.

Synopsys

show snmp stats - alias for 'stat snmp'

snmp trap

Sep 22, 2015

The following operations can be performed on "snmp trap":

add | rm | set | unset | show

add snmp trap

Adds an SNMP trap listener. You can configure the NetScaler appliance to generate asynchronous events (trap messages) to report abnormal conditions. The trap messages are sent to a remote device (trap listener) to help administrators monitor the appliance and respond promptly to any issues.

Synopsys

add snmp trap <trapClass> <trapDestination> ... [-version (V1 | V2)] [-destPort <port>] [-communityName <string>] [-srcIP <ip_addr|ipv6_addr>] [-severity <severity>]

Arguments

trapClass

Type of trap messages that the NetScaler appliance sends to the trap listener: Generic or the enterprise-specific messages defined in the MIB file.

Possible values: generic, specific

trapDestination

IPv4 or the IPv6 address of the trap listener to which the NetScaler appliance is to send SNMP trap messages.

version

SNMP version, which determines the format of trap messages sent to the trap listener.

This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Possible values: V1, V2

Default value: TRAP_VERSION_2

destPort

UDP port at which the trap listener listens for trap messages. This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Default value: 162

Minimum value: 1

Maximum value: 65534

communityName

Password (string) sent with the trap messages, so that the trap listener can authenticate them. Can include 1 to 31 uppercase or lowercase letters, numbers, and hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) characters.

You must specify the same community string on the trap listener device. Otherwise, the trap listener drops the trap messages.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

Default value: "public"

srcIP

IPv4 or IPv6 address that the NetScaler appliance inserts as the source IP address in all SNMP trap messages that it sends to this trap listener. By default this is the appliance's NSIP or NSIP6 address, but you can specify an IPv4 MIP or SNIP address or a SNIP6 address.

severity

Severity level at or above which the NetScaler appliance sends trap messages to this trap listener. The severity levels, in increasing order of severity, are Informational, Warning, Minor, Major, Critical. This parameter can be set for trap listeners of type SPECIFIC only. The default is to send all levels of trap messages.

Important: Trap messages are not assigned severity levels unless you specify severity levels when configuring SNMP alarms.

Possible values: Critical, Major, Minor, Warning, Informational

Default value: SNMP_SEV_UNKNOWN

---

rm snmp trap

Removes a trap listener entry from the NetScaler appliance.

Synopsys

rm snmp trap <trapClass> <trapDestination> ...

Arguments

trapClass

Trap type specified in the trap listener entry that you want to remove.

Possible values: generic, specific

trapDestination

IP address of the trap listener specified in the trap listener entry that you want to remove.

set snmp trap

Modifies the specified parameters in a trap-listener entry.

Synopsys

set snmp trap <trapClass> <trapDestination> [-destPort <port>] [-version (V1 | V2)] [-communityName <string>] [-srcIP <ip_addr|ipv6_addr>] [-severity <severity>]

Arguments

trapClass

Type of trap specified in the trap-listener entry. Because this parameter is used for identifying the trap listener entry, it cannot be modified after the entry has been created.

Possible values: generic, specific

trapDestination

IPv4 or the IPv6 address of the trap listener to which the NetScaler appliance is to send SNMP trap messages.

destPort

UDP port at which the trap listener listens for trap messages. This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Default value: 162

Minimum value: 1

Maximum value: 65534

version

SNMP version, which determines the format of trap messages sent to the trap listener.

This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Possible values: V1, V2

Default value: TRAP_VERSION_2

communityName

Password (string) sent with the trap messages, so that the trap listener can authenticate them. Can include 1 to 31 uppercase or lowercase letters, numbers, and hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) characters.

You must specify the same community string on the trap listener device. Otherwise, the trap listener drops the trap messages.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

Default value: "public"

srcIP

IPv4 or IPv6 address that the NetScaler appliance inserts as the source IP address in all SNMP trap messages that it sends to this trap listener. By default this is the appliance's NSIP or NSIP6 address, but you can specify an IPv4 MIP or SNIP address or a SNIP6 address.

severity

Severity level at or above which the NetScaler appliance sends trap messages to this trap listener. The severity levels, in increasing order of severity, are Informational, Warning, Minor, Major, Critical. This parameter can be set for trap listeners of type SPECIFIC only. The default is to send all levels of trap messages.

Important: Trap messages are not assigned severity levels unless you specify severity levels when configuring SNMP alarms.

Possible values: Critical, Major, Minor, Warning, Informational

Default value: SNMP_SEV_UNKNOWN

Example

```
set snmp trap generic 192.168.3.4 -version V1
```

---

unset snmp trap

Resets the specified parameters to their default settings in a trap-listener entry..Refer to the set snmp trap command for meanings of the arguments.

Synopsys

```
unset snmp trap <trapClass> <trapDestination> [-destPort] [-version] [-communityName] [-srcIP] [-severity]
```

Example

```
unset snmp trap generic 192.168.3.4 -version
```

---

show snmp trap

Displays the settings of all trap listeners or of the specified trap listener. To display the settings of all the trap listeners, run the command without any parameters. To display the settings of a particular trap listener, specify the trapClass (Trap Type) and trapDestination (IP Address) of the trap listener.

Synopsys

```
show snmp trap [<trapClass> <trapDestination>]
```

Arguments

trapClass

Trap type specified in the trap listener entry.

Possible values: generic, specific

summary

fullValues

format

level

Outputs

destPort

The destination port of the SNMP trap.

version

The SNMP version of the trap to be sent.

communityName

Password (string) sent with the trap messages, so that the trap listener can authenticate them. Can include 1 to 31 uppercase or lowercase letters, numbers, and hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore (_) characters.

You must specify the same community string on the trap listener device. Otherwise, the trap listener drops the trap messages.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

srcIP

The source IP of the SNMP trap to be sent.

severity

The minimum severity of traps to be sent to this destination.

devno

count

stateflag

Example

snmp user

Sep 22, 2015

The following operations can be performed on "snmp user":

add | rm | set | unset | show

add snmp user

Adds an SNMPv3 user who can send SNMP queries to the NetScaler appliance. You can add a maximum of 1000 SNMPv3 users.

Synopsys

add snmp user <name> -group <string> [-authType (MD5 | SHA) {-authPasswd } [-privType (DES | AES) {-privPasswd }]]

Arguments

name

Name for the SNMPv3 user. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my user" or 'my user').

group

Name of the configured SNMPv3 group to which to bind this SNMPv3 user. The access rights (bound SNMPv3 views) and security level set for this group are assigned to this user.

authType

Authentication algorithm used by the NetScaler appliance and the SNMPv3 user for authenticating the communication between them. You must specify the same authentication algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: MD5, SHA

privType

Encryption algorithm used by the NetScaler appliance and the SNMPv3 user for encrypting the communication between them. You must specify the same encryption algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: DES, AES

rm snmp user

Removes an SNMPv3 user entry from the NetScaler appliance.

Synopsys

rm snmp user <name>

Arguments

name

Name of the SNMPv3 user.

---

set snmp user

Modifies the specified parameters of an SNMPv3 user entry on the NetScaler appliance.

Synopsys

set snmp user <name> [-group <string>] [-authType (MD5 | SHA) {-authPasswd }] [-privType (DES | AES) {-privPasswd }]

Arguments

name

Name specified in the SNMPv3 user entry that you want to modify. Because this parameter is used for identifying the SNMPv3 user entry, it cannot be modified after the entry has been created.

group

Name of the configured SNMPv3 group to which to bind this SNMPv3 user. The access rights (bound SNMPv3 views) and security level set for this group are assigned to this user.

authType

Authentication algorithm used by the NetScaler appliance and the SNMPv3 user for authenticating the communication between them. You must specify the same authentication algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: MD5, SHA

privType

Encryption algorithm used by the NetScaler appliance and the SNMPv3 user for encrypting the communication between them. You must specify the same encryption algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: DES, AES

---

unset snmp user

Resets the specified parameters of an SNMPv3 user entry to their default settings..Refer to the set snmp user command for meanings of the arguments.

Synopsys

unset snmp user <name> (-authType | -privType) [-authPasswd] [-privPasswd]

---

show snmp user

Displays the settings of all SNMPv3 users or of the specified SNMPv3 user. To display the settings of all the SNMPv3 users, run the command without any parameters. To display the settings of a particular SNMPv3 user, specify the name of the SNMPv3 user.

Synopsys

show snmp user [<name>]

Arguments

name

Name of the SNMPv3 user whose details you want the NetScaler appliance to display.

summary

fullValues

format

level

Outputs

group

Name of the configured SNMPv3 group to which to bind this SNMPv3 user. The access rights (bound SNMPv3 views) and security level set for this group are assigned to this user.

authType

Authentication algorithm used by the NetScaler appliance and the SNMPv3 user for authenticating the communication between them. You must specify the same authentication algorithm when you configure the SNMPv3 user in the SNMP manager.

authPasswd

Plain-text pass phrase to be used by the authentication algorithm specified by the authType (Authentication Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the pass phrase includes one or more spaces, enclose it in double or single quotation marks (for example, "my phrase" or 'my phrase').

privType

Encryption algorithm used by the NetScaler appliance and the SNMPv3 user for encrypting the communication between them. You must specify the same encryption algorithm when you configure the SNMPv3 user in the SNMP manager.

privPasswd

Encryption key to be used by the encryption algorithm specified by the privType (Encryption Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the key includes one or more spaces, enclose it in double or single quotation marks (for example, "my key" or 'my key').

engineID

The context engine ID of the user.

storageType

The storage type for this user.

status

The status of this user.

devno

count

stateflag

snmp view

Sep 22, 2015

The following operations can be performed on "snmp view":

add | rm | set | show

add snmp view

Adds an SNMPv3 view. Used to implement access control for the SNMPv3 user, SNMPv3 views restrict user access to specific portions of the MIB. The NetScaler appliance can have multiple SNMPv3 views with the same name, differentiated by subtree parameter settings. You can add a maximum of 1000 SNMPv3 views.

Synopsys

add snmp view <name> <subtree> -type (included | excluded)

Arguments

name

Name for the SNMPv3 view. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the SNMPv3 view.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my view" or 'my view').

subtree

A particular branch (subtree) of the MIB tree that you want to associate with this SNMPv3 view. You must specify the subtree as an SNMP OID.

type

Include or exclude the subtree, specified by the subtree parameter, in or from this view. This setting can be useful when you have included a subtree, such as A, in an SNMPv3 view and you want to exclude a specific subtree of A, such as B, from the SNMPv3 view.

Possible values: included, excluded

---

rm snmp view

Removes an SNMPv3 view entry from the NetScaler appliance. The appliance can have multiple SNMPv3 views with the same name, differentiated by the subtree parameter setting. Therefore, to identify an SNMPv3 group subtree that you want to remove, you have to specify both the name and subtree of the SNMPv3 view.

Synopsys

rm snmp view <name> <subtree>

Arguments

name

Name of the SNMPv3 view. Note: If multiple views have the same name, specify the subtree to identify the view to be removed.

subtree

A MIB subtree of the SNMPv3 view.

---

set snmp view

Modifies the type (Type) parameter of an SNMPv3 view configured on the NetScaler appliance.

Synopsys

set snmp view <name> <subtree> -type (included | excluded)

Arguments

name

The name specified in the SNMPv3 view entry. This parameter cannot be modified.

subtree

A MIB subtree of the SNMPv3 view entry. This parameter cannot be modified.

type

Include or exclude the subtree, specified by the subtree parameter, in or from this view. This setting can be useful when you have included a subtree, such as A, in an SNMPv3 view and you want to exclude a specific subtree of A, such as B, from the SNMPv3 view.

Possible values: included, excluded

---

show snmp view

Displays the settings of all SNMPv3 views or of the specified SNMPv3 view. To display the settings of all the SNMPv3 views, run the command without any parameters. To display the settings of a particular SNMPv3 view, specify the name of the SNMPv3 view and subtree (the associated subtree of the MIB). The NetScaler appliance can have multiple SNMPv3 views with the same name, differentiated by the subtree parameter settings.

Synopsys

show snmp view [<name> [<subtree>]]

Arguments

name

Name of the SNMPv3 view.

summary

fullValues

format

level

Outputs

type

The type of sutree.

storageType

The storage type for this view.

status

The status of this view.

devno

count

stateflag

Spillover Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- spillover action
- spillover policy

spillover action

Sep 22, 2015

The following operations can be performed on "spillover action":

add | rm | show | rename

add spillover action

Creating spillover action

Synopsys

add spillover action <name> -action SPILLOVER

Arguments

name

Name of the spillover action.

action

Spillover action. Currently only type SPILLOVER is supported

Possible values: SPILLOVER

---

rm spillover action

Removes a spillover policy.

Synopsys

rm spillover action <name>

Arguments

name

Name of the spillover action.

---

show spillover action

Displaying spillover actions

Synopsys

show spillover action [<name>]

Arguments

name

Name of the spillover action.

summary

fullValues

format

level

Outputs

action

Spillover action. Currently only type SPILLOVER is supported

builtin

Flag to determine whether compression is default or not

devno

count

stateflag

---

rename spillover action

Renames a spillover action.

Synopsys

rename spillover action <name>@ <newName>@

Arguments

name

Existing name of the action.

newName

New name for the spillover action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

Choose a name that can be correlated with the function that the action performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

Example

```
rename spillover policy oldname newname
```

spillover policy

Sep 22, 2015

The following operations can be performed on "spillover policy":

add | rm | set | unset | show | rename | stat

add spillover policy

Add a spillover policy. SPILLOVER policies that can be added are based on vserver expressions.

Synopsys

add spillover policy <name> -rule <expression> -action <string> [-comment <string>]

Arguments

name

Name of the spillover policy.

rule

Expression to be used by the spillover policy.

action

Action for the spillover policy. Action is created using add spillover action command

comment

Any comments that you might want to associate with the spillover policy.

Example

add spillover policy pol1 -rule "SYS.VSERVER("abc").ACTIVESERVICES.LE(2) -action act1 add spillover policy pol2 -rule "SYS.VSERVER("abc").CONNECTIONS.GT(500) -ac

---

rm spillover policy

Removes a spillover policy.

Synopsys

rm spillover policy <name>

Arguments

name

Name of the spillover policy.

---

set spillover policy

Used to change the expression or other parameters of an existing policy.

Synopsys

set spillover policy <name> [-rule <expression>] [-action <string>] [-comment <string>]

Arguments

name

Name of the spillover policy.

rule

Expression to be used by the spillover policy.

action

Action for the spillover policy. Action is created using add spillover action command

comment

Any comments that you might want to associate with the spillover policy.

Example

set spillover policy pol1 -rule "SYS.VSERVER("abc").ACTIVESERVICS.LE(1)" set spillover policy pol2 -action act4"

---

unset spillover policy

Use this command to remove spillover policy settings.Refer to the set spillover policy command for meanings of the arguments.

Synopsys

unset spillover policy <name> -comment

---

show spillover policy

Displaying the policy-related information.

Synopsys

show spillover policy [<name>]

Arguments

name

Name of the spillover policy.

summary

fullValues

format

level

Outputs

stateflag

rule

Expression to be used by the spillover policy.

action

Action for the spillover policy. Action is created using add spillover action command

boundTo

The name of the entity to which the policy is bound.

hits

The number of times the policy has been hit.

undefHits

Number of policy UNDEF hits.

activePolicy

Indicates whether policy is bound or not.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any comments that you might want to associate with the spillover policy.

builtin

Flag to determine if compression policy is builtin or not

devno

count

---

rename spillover policy

Renames a spillover policy.

Synopsys

rename spillover policy <name>@ <newName>@

Arguments

name

Existing name of the policy.

newName

New name for the spillover policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Choose a name that reflects the function that the policy performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

Example

rename spillover policy oldname newname

---

stat spillover policy

Displays statistics for all spillover policies currently configured on the NetScaler appliance, or detailed statistics for the specified policy.

Synopsys

stat spillover policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the spillover policy for which to show detailed statistics.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

SSL Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- ssl
- ssl action
- ssl cert
- ssl certKey
- ssl certLink
- ssl certReq
- ssl cipher
- ssl ciphersuite
- ssl crl
- ssl dhParam
- ssl dsaKey
- ssl dtlsProfile
- ssl fips
- ssl fipsKey
- ssl fipsSIMSource
- ssl fipsSIMTarget
- ssl global
- ssl ocspResponder
- ssl parameter
- ssl pkcs12
- ssl pkcs8
- ssl policy
- ssl policylabel
- ssl rsakey
- ssl service
- ssl serviceGroup
- ssl stats
- ssl vserver
- ssl wrapkey

ssl

Sep 22, 2015

The following operations can be performed on "ssl":

stat ssl

Displays SSL statistics.

Synopsys

stat ssl [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

 # SSL cards UP (SSLCardUP)

Number of SSL cards that are UP. If the number of cards UP is lower than a threshold, a failover is initiated.

SSL crypto card status (SSLCardSt)

Status of the SSL card(s). The value should be interpreted in binary form, with each set bit indicates a card as UP.

 # SSL cards present (SSLCards)

Number of SSL crypto cards present on the NetScaler appliance.

SSL engine status (SSLEngSt)

State of the SSL Engine (1=UP/0=DOWN). This state is decided based on SSL Feature/License status and minimum number of cards UP.

SSL sessions (SSLSe)

Number of SSL sessions on the NetScaler appliance.

SSL transactions (SSLTrn)

Number of SSL transactions on the NetScaler appliance.

SSLv2 transactions (SSL2Trn)

Number of SSLv2 transactions on the NetScaler appliance.

SSLv3 transactions (SSL3Trn)

Total number of SSLv3 transactions on the NetScaler appliance.

TLSv1 transactions (TLS1Trn)

Number of TLSv1 transactions on the NetScaler appliance.

SSLv2 sessions (SSL2Se)

Number of SSLv2 sessions on the NetScaler appliance.

SSLv3 sessions (SSL3Se)

Number of SSLv3 sessions on the NetScaler appliance.

TLSv1 sessions (TLS1Se)

Number of TLSv1 sessions on the NetScaler appliance.

new SSL sessions (NewSe)

Number of new SSL sessions created on the NetScaler appliance.

SSL session misses (SeMiss)

Number of SSL session reuse misses on the NetScaler appliance.

SSL session hits (SeHit)

Number of SSL session reuse hits on the NetScaler appliance.

SSL sessions (BSSLSe)

Number of back-end SSL sessions on the NetScaler appliance.

SSLv3 sessions (BSSL3Se)

Number of back-end SSLv3 sessions on the NetScaler appliance.

TLSv1 sessions (BTLS1Se)

Number of back-end TLSv1 sessions on the NetScaler appliance.

Session multiplex attempts (BSeMx)

Number of back-end SSL session multiplex attempts on the NetScaler appliance.

Session multiplex successes (BSeMxS)

Number of back-end SSL session multiplex successes on the NetScaler appliance.

Session multiplex failures (BSeMxF)

Number of back-end SSL session multiplex failures on the NetScaler appliance.

Bytes encrypted (Enc)

Number of bytes encrypted on the NetScaler appliance.

Bytes decrypted (Dec)

Number of bytes decrypted on the NetScaler appliance.

SSL session renegotiations (SSLRn)

Number of SSL session renegotiations on the NetScaler appliance.

SSLv3 session renegotiations (SSL3Rn)

Number of session renegotiations done on SSLv3.

TLSv1 session renegotiations (TLS1Rn)

Number of SSL session renegotiations done on TLSv1.

RSA 512-bit key exchanges (RSAKx5)

Number of RSA 512-bit key exchanges on the NetScaler appliance.

RSA 1024-bit key exchanges (RSAKx1)

Number of RSA 1024-bit key exchanges on the NetScaler appliance.

RSA 2048-bit key exchanges (RSAKx2)

Number of RSA 2048-bit key exchanges on the NetScaler appliance.

RSA 4096-bit key exchanges (RSAKx4)

Number of RSA 4096-bit key exchanges on the NetScaler appliance.

DH 512-bit key exchanges (DHKx5)

Number of Diffie-Helman 512-bit key exchanges on the NetScaler appliance.

DH 1024-bit key exchanges (DHKx1)

Number of Diffie-Helman 1024-bit key exchanges on the NetScaler appliance.

DH 2048-bit key exchanges (DHKx2)

Number of Diffie-Helman 2048-bit key exchanges on the NetScaler appliance.

ECDHE 521 curve key exchanges (ECDHEKx521)

Number of 521 Elliptical Curve Diffie-Helman on the NetScaler appliance.

ECDHE 384 curve key exchanges (ECDHEKx384)

Number of 384 Elliptical Curve Diffie-Helman on the NetScaler appliance.

ECDHE 256 curve key exchanges (ECDHEKx256)

Number of 256 Elliptical Curve Diffie-Helman on the NetScaler appliance.

ECDHE 224 curve key exchanges (ECDHEKx224)

Number of 224 Elliptical Curve Diffie-Helman on the NetScaler appliance.

RC4 40-bit encryptions (RC4En4)

Number of RC4 40-bit cipher encryptions on the NetScaler appliance.

RC4 56-bit encryptions (RC4En5)

Number of RC4 56-bit cipher encryptions on the NetScaler appliance.

RC4 64-bit encryptions (RC4En6)

Number of RC4 64-bit cipher encryptions on the NetScaler appliance.

RC4 128-bit encryptions (RC4En1)

Number of RC4 128-bit cipher encryptions on the NetScaler appliance.

DES 40-bit encryptions (DESEn4)

Number of DES 40-bit cipher encryptions on the NetScaler appliance.

DES 56-bit encryptions (DESEn5)

Number of DES 56-bit cipher encryptions on the NetScaler appliance.

3DES 168-bit encryptions (3DESEn1)

Number of DES 168-bit cipher encryptions on the NetScaler appliance.

AES 128-bit encryptions (AESEn1)

Number of AES 128-bit cipher encryptions on the NetScaler appliance.

AES 256-bit encryptions (AESEn2)

Number of AES 256-bit cipher encryptions on the NetScaler appliance.

RC2 40-bit encryptions (RC2En4)

Number of RC2 40-bit cipher encryptions on the NetScaler appliance.

RC2 56-bit encryptions (RC2En5)

Number of RC2 56-bit cipher encryptions on the NetScaler appliance.

RC2 128-bit encryptions (RC2En1)

Number of RC2 128-bit cipher encryptions on the NetScaler appliance.

Null cipher encryptions (NullEn)

Number of Null cipher encryptions on the NetScaler appliance.

MD5 hashes (MD5Hsh)

Number of MD5 hashes on the NetScaler appliance.

SHA hashes (SHAHsh)

Number of SHA hashes on the NetScaler appliance.

SSLv2 SSL handshakes (SSL2Hs)

Number of handshakes on SSLv2 on the NetScaler appliance.

SSLv3 SSL handshakes (SSL3Hs)

Number of handshakes on SSLv3 on the NetScaler appliance.

TLSv1 SSL handshakes (TLS1Hs)

Number of SSL handshakes on TLSv1 on the NetScaler appliance.

SSLv2 client authentications (SSL2CAt)

Number of client authentications done on SSLv2.

SSLv3 client authentications (SSL3CAt)

Number of client authentications done on SSLv3.

TLSv1 client authentications (TLS1CAt)

Number of client authentications done on TLSv1.

RSA authentications (RSAAt)

Number of RSA authentications on the NetScaler appliance.

DH authentications (DHAt)

Number of Diffie-Helman authentications on the NetScaler appliance.

DSS (DSA) authentications (DSSAt)

Total number of times DSS authorization is used on the NetScaler appliance.

Null authentications (NullAt)

Number of Null authentications on the NetScaler appliance.

SSL session renegotiations (BSSLRn)

Number of back-end SSL session renegotiations on the NetScaler appliance.

SSLv3 session renegotiations (BSSL3Rn)

Number of back-end SSLv3 session renegotiations on the NetScaler appliance.

TLSv1 session renegotiations (BTLS1Rn)

Number of back-end TLSv1 session renegotiations on the NetScaler appliance.

RSA 512-bit key exchanges (BRSAKx5)

Number of back-end RSA 512-bit key exchanges on the NetScaler appliance.

RSA 1024-bit key exchanges (BRSAKx1)

Number of back-end RSA 1024-bit key exchanges on the NetScaler appliance.

RSA 2048-bit key exchanges (BRSAKx2)

Number of back-end RSA 2048-bit key exchanges on the NetScaler appliance.

DH 512-bit key exchanges (BDHKx5)

Number of back-end DH 512-bit key exchanges on the NetScaler appliance.

DH 1024-bit key exchanges (BDHKx1)

Number of back-end DH 1024-bit key exchanges on the NetScaler appliance.

DH 2048-bit key exchanges (BDHKx2)

Number of back-end DH 2048-bit key exchanges on the NetScaler appliance.

RC4 40-bit encryptions (BRC4En4)

Number of back-end RC4 40-bit cipher encryptions on the NetScaler appliance.

RC4 56-bit encryptions (BRC4En5)

Number of back-end RC4 56-bit cipher encryptions on the NetScaler appliance.

RC4 64-bit encryptions (BRC4En6)

Number of back-end RC4 64-bit cipher encryptions on the NetScaler appliance.

RC4 128-bit encryptions (BRC4En1)

Number of back-end RC4 128-bit cipher encryptions on the NetScaler appliance.

DES 40-bit encryptions (BDESEn4)

Number of back-end DES 40-bit cipher encryptions on the NetScaler appliance.

DES 56-bit encryptions (BDESEn5)

Number of back-end DES 56-bit cipher encryptions on the NetScaler appliance.

3DES 168-bit encryptions (B3DESE1n)

Number of back-end 3DES 168-bit cipher encryptions on the NetScaler appliance.

AES 128-bit encryptions (BAESEn1)

Back-end AES 128-bit cipher encryptions on the NetScaler appliance.

AES 256-bit encryptions (BAESEn2)

Back-end AES 256-bit cipher encryptions on the NetScaler appliance.

RC2 40-bit encryptions (BRC2En4)

Number of back-end RC2 40-bit cipher encryptions on the NetScaler appliance.

RC2 56-bit encryptions (BRC2En5)

Number of back-end RC2 56-bit cipher encryptions on the NetScaler appliance.

RC2 128-bit encryptions (BRC2En1)

Number of back-end RC2 128-bit cipher encryptions on the NetScaler appliance.

null encryptions (BNullEn)

Number of back-end null cipher encryptions on the NetScaler appliance.

MD5 hashes (BMD5Hsh)

Number of back-end MD5 hashes on the NetScaler appliance.

SHA hashes (BSHAHsh)

Number of back-end SHA hashes on the NetScaler appliance.

SSLv3 handshakes (BSSL3Hs)

Number of back-end SSLv3 handshakes on the NetScaler appliance.

TLSv1 handshakes (BTLS1Hs)

Number of back-end TLSv1 handshakes on the NetScaler appliance.

SSLv3 client authentications (BSSL3CAt)

Number of back-end SSLv3 client authentications on the NetScaler appliance.

TLSv1 client authentications (BTLS1CAt)

Number of back-end TLSv1 client authentications on the NetScaler appliance.

RSA authentications (BRSAAt)

Number of back-end RSA authentications on the NetScaler appliance.

DH authentications (BDHAt)

Number of back-end DH authentications on the NetScaler appliance.

DSS authentications (BDSSAt)

Number of back-end DSS authentications on the NetScaler appliance.

Null authentications (BNullAt)

Number of back-end null authentications on the NetScaler appliance.

RSA key exchanges offloaded (RSAkxOf)

Number of RSA key exchanges offloaded to the cryptography card.

RSA sign operations offloaded (RSASnOf)

Number of RSA sign operations offloaded to the cryptography card.

DH key exchanges offloaded (DHkxOf)

Number of DH key exchanges offloaded to the cryptography card.d

RC4 encryptions offloaded (RC4EnOf)

Number of RC4 encryptions offloaded to the cryptography card.

DES encryptions offloaded (DESEnOf)

Number of DES encryptions offloaded to the cryptography card.

AES encryptions offloaded (AESEnOf)

Number of AES encryptions offloaded to the cryptography card.

Bytes encrypted in hardware (EncHw)

Number of bytes encrypted in hardware.

Bytes encrypted in software. (EncSw)

Number of bytes encrypted in software.

Bytes encrypted on the front end. (EncFe)

Number of bytes encrypted on the front end.

Bytes encrypted in hardware on the front end. (EncHwFe)

Number of bytes encrypted in hardware on the front end.

Bytes encrypted in software on front-end (EncSwFe)

Number of bytes encrypted in software on the front end.

Bytes encrypted on back-end (EncBe)

Number of bytes encrypted on the back end.

Bytes encrypted in hardware on back-end (EncHwBe)

Number of bytes encrypted in hardware on the back end.

Bytes encrypted in software on back-end (EncSwBe)

Number of bytes encrypted in software on the back end.

Bytes decrypted in hardware (DecHw)

Number of bytes decrypted in hardware.

Bytes decrypted in software (DecSw)

Number of bytes decrypted in software.

Bytes decrypted on front-end (DecFe)

Number of bytes decrypted on the front end.

Bytes decrypted in hardware on front-end (DecHwFe)

Number of bytes decrypted in hardware on the front end.

Bytes decrypted in software on front-end (DecSwFe)

Number of bytes decrypted in software on the front end.

Bytes decrypted on back-end (DecBe)

Number of bytes decrypted on the back end.

Bytes decrypted in hardware on back-end (DecHwBe)

Number of bytes decrypted in hardware on the back end.

Bytes decrypted in software on the back end. (DecSwBe)

Number of bytes decrypted in software on back-end

Backend SSL sessions reused (BSeRe)

Number of back-end SSL sessions reused on the NetScaler appliance.

IDEA 128-bit encryptions (IDEAEn1)

Number of IDEA 128-bit cipher encryptions on the NetScaler appliance.

IDEA 128-bit encryptions (BIDEAEn1)

Number of back-end IDEA 128-bit cipher encryptions on the NetScaler appliance.

ssl action

Sep 22, 2015

The following operations can be performed on "ssl action":

add | rm | show

add ssl action

Creates a new SSL action. An SSL action defines SSL settings that you can apply to the selected requests. You associate an action with one or more policies. Data in client connection requests or responses is compared to a rule (expression) specified in the policy, and the action is applied to connections that match the rule.

Synopsys

add ssl action <name> [-clientAuth (DOCLIENTAUTH | NOCLIENTAUTH)] [-clientCert (ENABLED | DISABLED) -certHeader <string>] [-clientCertSerialNumber (ENABLED | DISABLED) -certSerialHeader <string>] [-clientCertSubject (ENABLED | DISABLED) -certSubjectHeader <string>] [-clientCertHash (ENABLED | DISABLED) -certHashHeader <string>] [-clientCertIssuer (ENABLED | DISABLED) -certIssuerHeader <string>] [-sessionID (ENABLED | DISABLED) -sessionIDHeader <string>] [-cipher (ENABLED | DISABLED) -cipherHeader <string>] [-clientCertNotBefore (ENABLED | DISABLED) -certNotBeforeHeader <string>] [-clientCertNotAfter (ENABLED | DISABLED) -certNotAfterHeader <string>] [-OWASupport (ENABLED | DISABLED)]

Arguments

name

Name for the SSL action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

clientAuth

Perform client certificate authentication.

Possible values: DOCLIENTAUTH, NOCLIENTAUTH

clientCert

Insert the entire client certificate into the HTTP header of the request being sent to the web server. The certificate is inserted in ASCII (PEM) format.

Possible values: ENABLED, DISABLED

clientCertSerialNumber

Insert the entire client serial number into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

clientCertSubject

Insert the client certificate subject, also known as the distinguished name (DN), into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

clientCertHash

Insert the certificate signature (hash) into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

clientCertIssuer

Insert the certificate issuer details into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

sessionID

Insert the SSL session ID into the HTTP header of the request being sent to the web server. Every SSL connection that the client and the NetScaler share has a unique ID that identifies the specific connection.

Possible values: ENABLED, DISABLED

cipher

Insert the cipher suite that the client and the NetScaler appliance negotiated for the SSL session into the HTTP header of the request being sent to the web server. The appliance inserts the cipher-suite name, SSL protocol, export or non-export string, and cipher strength bit, depending on the type of browser connecting to the SSL virtual server or service (for example, Cipher-Suite: RC4- MD5 SSLv3 Non-Export 128-bit).

Possible values: ENABLED, DISABLED

clientCertNotBefore

Insert the date from which the certificate is valid into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time from which it is valid.

Possible values: ENABLED, DISABLED

clientCertNotAfter

Insert the date of expiry of the certificate into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time at which the certificate expires.

Possible values: ENABLED, DISABLED

OWASupport

If the appliance is in front of an Outlook Web Access (OWA) server, insert a special header field, FRONT-END-HTTPS: ON, into the HTTP requests going to the OWA server. This header communicates to the server that the transaction is HTTPS and not HTTP.

Possible values: ENABLED, DISABLED

Example

```
add ssl action certInsert_act -clientCert ENABLED -certHeader CERT
```

---

rm ssl action

Removes the specified SSL action.

Synopsys

```
rm ssl action <name>
```

Arguments

name

Name of the SSL action to remove.

Example

```
rm ssl action certInsert_act
```

---

show ssl action

Displays information about all the SSL actions configured on the appliance, or displays detailed information about the specified SSL action.

Synopsys

```
show ssl action [<name>]
```

Arguments

name

Name of the SSL action for which to show detailed information.

summary

fullValues

format

level

Outputs

stateflag

clientAuth

Perform client certificate authentication.

clientCert

Insert the entire client certificate into the HTTP header of the request being sent to the web server. The certificate is inserted in ASCII (PEM) format.

certHeader

clientCertSerialNumber

Insert the entire client serial number into the HTTP header of the request being sent to the web server.

certSerialHeader

clientCertSubject

Insert the client certificate subject, also known as the distinguished name (DN), into the HTTP header of the request being sent to the web server.

certSubjectHeader

clientCertHash

Insert the certificate signature (hash) into the HTTP header of the request being sent to the web server.

certHashHeader

clientCertIssuer

Insert the certificate issuer details into the HTTP header of the request being sent to the web server.

certIssuerHeader

sessionID

Insert the SSL session ID into the HTTP header of the request being sent to the web server. Every SSL connection that the client and the NetScaler share has a unique ID that identifies the specific connection.

sessionIDHeader

cipher

Insert the cipher suite that the client and the NetScaler appliance negotiated for the SSL session into the HTTP header of the request being sent to the web server. The appliance inserts the cipher-suite name, SSL protocol, export or non-export string, and cipher strength bit, depending on the type of browser connecting to the SSL virtual server or service (for example, Cipher-Suite: RC4- MD5 SSLv3 Non-Export 128-bit).

cipherHeader

OWASupport

If the appliance is in front of an Outlook Web Access (OWA) server, insert a special header field, FRONT-END-HTTPS: ON, into the HTTP requests going to the OWA server. This header communicates to the server that the transaction is HTTPS and not HTTP.

clientCertNotBefore

Insert the date from which the certificate is valid into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time from which it is valid.

certNotBeforeHeader

clientCertNotAfter

Insert the date of expiry of the certificate into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time at which the certificate expires.

certNotAfterHeader

hits

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

description

Description of the action

flags

builtin

Flag to determine whether ssl action is built-in or not

devno

count

Example

show ssl action 1 Configured SSL action: 1) Name: certIns

ssl cert

Sep 22, 2015

The following operations can be performed on "ssl cert":

create ssl cert

Generates a signed X509 Certificate.

Synopsys

create ssl cert <certFile> <reqFile> <certType> [-keyFile <input_filename>] [-keyform (DER | PEM) {-PEMPassPhrase }] [-days <positive_integer>] [-certForm (DER | PEM)] [-CAcert <input_filename>] [-CAcertForm (DER | PEM)] [-CAkey <input_filename>] [-CAkeyForm (DER | PEM)] [-CAserial <output_filename>]

Arguments

certFile

Name for and, optionally, path to the generated certificate file. /nsconfig/ssl/ is the default path.

Maximum value: 63

reqFile

Name for and, optionally, path to the certificate-signing request (CSR). /nsconfig/ssl/ is the default path.

Maximum value: 63

certType

Type of certificate to generate. Specify one of the following:

* ROOT_CERT - Self-signed Root-CA certificate. You must specify the key file name. The generated Root-CA certificate can be used for signing end-user client or server certificates or to create Intermediate-CA certificates.

* INTM_CERT - Intermediate-CA certificate.

* CLNT_CERT - End-user client certificate used for client authentication.

* SRVR_CERT - SSL server certificate used on SSL servers for end-to-end encryption.

Possible values: ROOT_CERT, INTM_CERT, CLNT_CERT, SRVR_CERT

keyFile

Name for and, optionally, path to the private key. You can either use an existing RSA or DSA key that you own or create a new private key on the NetScaler appliance. This file is required only when creating a self-signed Root-CA certificate. The key file is stored in the /nsconfig/ssl directory by default.

If the input key specified is an encrypted key, you are prompted to enter the PEM pass phrase that was used for encrypting the key.

Maximum value: 63

keyform

Format in which the key is stored on the appliance.

Possible values: DER, PEM

Default value: FORMAT_PEM

PEMPassPhrase

days

Number of days for which the certificate will be valid, beginning with the time and day (system time) of creation.

Default value: 365

Minimum value: 1

Maximum value: 3650

certForm

Format in which the certificate is stored on the appliance.

Possible values: DER, PEM

Default value: FORMAT_PEM

CAcert

Name of the CA certificate file that issues and signs the Intermediate-CA certificate or the end-user client and server certificates.

Maximum value: 63

CAcertForm

Format of the CA certificate.

Possible values: DER, PEM

Default value: FORMAT_PEM

CAkey

Private key, associated with the CA certificate that is used to sign the Intermediate-CA certificate or the end-user client and server certificate. If the CA key file is password protected, the user is prompted to enter the pass phrase that was used to encrypt the key.

Maximum value: 63

CAkeyForm

Format for the CA certificate.

Possible values: DER, PEM

Default value: FORMAT_PEM

CAserial

Serial number file maintained for the CA certificate. This file contains the serial number of the next certificate to be issued or signed by the CA. If the specified file does not exist, a new file is created, with /nsconfig/ssl/ as the default path. If you do not specify a proper path for the existing serial file, a new serial file is created. This might change the certificate serial numbers assigned by the CA certificate to each of the certificates it signs.

Maximum value: 63

Example

1) create ssl cert /nsconfig/ssl/root_cert.pem /nsconfig/ssl/root_csr.pem ROOT_CERT -keyFile /nsconfig/ssl/root_key.pem -days 1000 The above example creates a self sign

ssl certKey

Sep 22, 2015

The following operations can be performed on "ssl certKey":

add | rm | set | unset | bind | unbind | link | unlink | show | update

add ssl certKey

Adds a certificate-key pair to memory. After it is bound to a virtual server or service, it is used for processing SSL transactions. In a high-availability configuration, the path to the certificate and the optional private key must be the same on the primary and the secondary appliance. For a server certificate, a private key is required.

Synopsys

add ssl certKey <certkeyName> -cert <string> [(-key <string> [-password]) | -fipsKey <string>] [-inform (DER | PEM)] [-expiryMonitor (ENABLED | DISABLED) [-notificationPeriod <positive_integer>]] [-bundle (YES | NO)]

Arguments

certkeyName

Name for the certificate and private-key pair. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the certificate-key pair is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cert" or 'my cert').

cert

Name of and, optionally, path to the X509 certificate file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. /nsconfig/ssl/ is the default path.

key

Name of and, optionally, path to the private-key file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. /nsconfig/ssl/ is the default path.

fipsKey

Name of the FIPS key that was created inside the Hardware Security Module (HSM) of a FIPS appliance, or a key that was imported into the HSM.

inform

Input format of the certificate and the private-key files. The two formats supported by the appliance are:

PEM - Privacy Enhanced Mail

DER - Distinguished Encoding Rule

Possible values: DER, PEM

Default value: FORMAT_PEM

passplain

Pass phrase used to encrypt the private-key. Required when adding an encrypted private-key in PEM format.

expiryMonitor

Issue an alert when the certificate is about to expire.

Possible values: ENABLED, DISABLED

notificationPeriod

Time, in number of days, before certificate expiration, at which to generate an alert that the certificate is about to expire.

Minimum value: 10

Maximum value: 100

bundle

Parse the certificate chain as a single file after linking the server certificate to its issuer's certificate within the file.

Possible values: YES, NO

Default value: NO

Example

1) add ssl certkey siteAcertkey -cert /nsconfig/ssl/cert.pem -key /nsconfig/ssl/pkey.pem The above command loads a certificate and private key file. 2) add ssl certkey site.

---

rm ssl certKey

Removes all the certificate-key pairs, or the specified certificate-key pair, from the appliance. The certificate-key pair is removed only if it is not referenced by any other object. The reference count is updated when the certificate-key pair is bound to an SSL virtual server or linked to another certificate-key pair.

Synopsys

rm ssl certKey <certkeyName> ...

Arguments

certkeyName

Name of the certificate-key pair to remove.

Example

1) rm ssl certkey siteAcertkey The above command removes the certificate-key pair siteAcertkey from the system.

---

set ssl certKey

Modifies the specified attributes of a certificate-key pair.

Synopsys

set ssl certKey <certkeyName> [-expiryMonitor (ENABLED | DISABLED) [-notificationPeriod <positive_integer>]]

Arguments

certkeyName

Name of the certificate-key pair to modify.

expiryMonitor

Issue an alert when the certificate is about to expire.

Possible values: ENABLED, DISABLED

---

unset ssl certKey

Use this command to remove ssl certKey settings.Refer to the set ssl certKey command for meanings of the arguments.

Synopsys

unset ssl certKey <certkeyName> [-expiryMonitor] [-notificationPeriod]

---

bind ssl certKey

Binds a certificate-key pair to an SSL virtual server or an SSL service.

Synopsys

bind ssl certKey [<certkeyName>] [-ocspResponder <string>] [-priority <positive_integer>]

Arguments

certkeyName

Name of the certificate-key pair.

ocspResponder

Name of the OCSP responder to be associated with the CA certificate.

vServerName

The name of the SSL virtual server name to which the certificate-key pair needs to be bound.

serviceName

The name of the SSL service to which the certificate-key pair needs to be bound. Use the ###add service### command to create this service.

serviceGroupName

The name of the SSL service group to which the certificate-key pair needs to be bound. Use the "add servicegroup" command to create this service.

CA

If this option is specified, it indicates that the certificate-key pair being bound to the SSL virtual server is a CA certificate. If this option is not specified, the certificate-key pair is bound as a normal server certificate.

Note: In case of a normal server certificate, the certificate-key pair should consist of both the certificate and the private-key.

Example

1) bind ssl certkey cacert -ocspResponder ocsp_ca -priority 1 In the above example, the CA certificate cacert is bound with the OCSP responder ocsp_ca with priority 1, wh

---

unbind ssl certKey

Unbinds the specified certificate-key pair from the SSL virtual server or service.

Synopsys

unbind ssl certKey <certkeyName> -ocspResponder <string>

Arguments

certkeyName

Name of the certificate-key pair to unbind.

ocspResponder

Name of the OCSP responder.

vServerName

The name of the SSL virtual server.

serviceName

The name of the SSL service

serviceGroupName

The name of the service group.

CA

The certificate-key pair being unbound is a Certificate Authority (CA) certificate. If you choose this option, the certificate-key pair is unbound from the list of CA certificates that were bound to the specified SSL virtual server or SSL service.

Example

1) unbind ssl certkey sslvip siteAcertkey In the above example, the server certificate siteAcertkey is unbound from the SSL virtual server. 2) unbind ssl certkey sslvip CAc

---

link ssl certKey

Links a certificate-key pair to its Certificate Authority (CA) certificate-key pair.

Synopsys

link ssl certKey <certkeyName> <linkCertKeyName>

Arguments

certkeyName

Name of the certificate-key pair to link to its issuer's certificate-key pair in the chain.

linkCertKeyName

Name of the Certificate Authority certificate-key pair to which to link a certificate-key pair.

Example

1) link ssl certkey siteAcertkey CAcertkey In the above example, the certificate-key siteAcertkey is bound to its issuer certificate-key pair CAcertkey.

---

unlink ssl certKey

Unlinks the certificate-key pair from its Certificate-Authority (CA) certificate-key pair.

Synopsys

unlink ssl certKey <certkeyName>

Arguments

certkeyName

Name of the certificate-key pair to unlink.

Example

1) unlink ssl certkey siteAcertkey The above example unlinks the certificate 'siteAcertkey' from its Certificate-Authority (CA) certificate.

---

show ssl certKey

Displays information about all the certificate-key pairs configured on the appliance, or displays detailed information about the specified certificate-key pair.

Synopsys

show ssl certKey [<certkeyName>]

Arguments

certkeyName

Name of the certificate-key pair for which to show detailed information.

summary

fullValues

format

level

Outputs

cert

The name and location of the file containing the certificate.

key

The name and location of the file containing the key.

inform

The encoding format of the certificate and key (PEM or DER).

signatureAlg

Signature algorithm.

serial

Serial number.

issuer

Issuer name.

clientCertNotBefore

Not-Before date.

clientCertNotAfter

Not-After date.

daysToExpiration

Days remaining for the certificate to expire.

subject

Subject name.

publickey

Public key algorithm.

publickeysize

Size of the public key.

version

Version.

priority

ocsp priority

status

Status of the certificate.

fipsKey

FIPS key ID.

passcrypt

Passcrypt.

data

Vserver Id

serverName

Vserver name to which the certificate key pair is bound.

serviceName

Service name to which the certificate key pair is bound.

ocspResponder

OCSP responders bound to this certkey

expiryMonitor

Certificate expiry monitor

notificationPeriod

Certificate expiry notification period

linkCertKeyName

The name of the Certificate-Authority.

stateflag

devno

count

Example

1) An example of the output of the show ssl certkey command is shown below: 2 configured certkeys: 1) Name: siteAcertkey Cert Path

---

update ssl certKey

Updates the certificate or private key in a certificate-key pair. In a high availability configuration, the path to the certificate and the optional private key must be the same on the primary and secondary nodes.

Synopsys

update ssl certKey <certkeyName> [-cert <string>] [(-key <string> [-password]) | -fipsKey <string>] [-inform (DER | PEM)] [-noDomainCheck]

Arguments

certkeyName

Name of the certificate-key pair to update.

cert

Name of and, optionally, path to the X509 certificate file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. /nsconfig/ssl/ is the default path.

key

Name of and, optionally, path to the private-key file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. /nsconfig/ssl/ is the default path.

fipsKey

Name of the FIPS key that was created inside the Hardware Security Module (HSM) of a FIPS appliance, or a key that was imported into the HSM.

inform

Input format of the certificate and the private-key files. The two formats supported by the appliance are:

PEM - Privacy Enhanced Mail

DER - Distinguished Encoding Rule

Possible values: DER, PEM

Default value: FORMAT_PEM

passplain

Pass phrase used to encrypt the private-key. Required when adding an encrypted private-key in PEM format.

noDomainCheck

Override the check for matching domain names during a certificate update operation.

Example

1) update ssl certkey siteAcertkey -cert /nsconfig/ssl/cert.pem -key /nsconfig/ssl/pkey.pem The above command updates a certificate

ssl certLink

Sep 22, 2015

The following operations can be performed on "ssl certLink":

show ssl certLink

Displays information about all the linked certificate-key pairs on the appliance.

Synopsys

show ssl certLink

Arguments

summary

fullValues

format

level

Outputs

certkeyName

Certificate key name.

linkCertKeyName

Name of the Certificate-Authority.

devno

count

stateflag

Example

The following shows an example of the output of the show ssl certlink command: linked certificate: 1) Cert Name: siteAcertkey CA Cer

ssl certReq

Sep 22, 2015

The following operations can be performed on "ssl certReq":

create ssl certReq

Generates a new Certificate Signing Request (CSR). A CSR is a collection of information including the domain name, company details, and the private key to be used to create the certificate. Send the CSR to a Certificate Authority (CA) to obtain an X509 certificate for the user domain (web site).

Synopsys

create ssl certReq <reqFile> (-keyFile <input_filename> | -fipsKeyName <string>) [-keyform (DER | PEM) {-PEMPassPhrase }] -countryName <string> -stateName <string> -organizationName <string> [-organizationUnitName <string>] [-localityName <string>] [-commonName <string>] [-emailAddress <string>] {-challengePassword } [-companyName <string>]

Arguments

reqFile

Name for and, optionally, path to the certificate signing request (CSR). /nsconfig/ssl/ is the default path.

Maximum value: 63

keyFile

Name of and, optionally, path to the private key used to create the certificate signing request, which then becomes part of the certificate-key pair. The private key can be either an RSA or a DSA key. The key must be present in the appliance's local storage. /nsconfig/ssl is the default path.

Maximum value: 63

fipsKeyName

Name of the FIPS key used to create the certificate signing request. FIPS keys are created inside the Hardware Security Module of the FIPS card.

keyform

Format in which the key is stored on the appliance.

Possible values: DER, PEM

Default value: FORMAT_PEM

countryName

Two letter ISO code for your country. For example, US for United States.

stateName

Full name of the state or province where your organization is located.

Do not abbreviate.

organizationName

Name of the organization that will use this certificate. The organization name (corporation, limited partnership, university, or government agency) must be registered with some authority at the national, state, or city level. Use the legal name under which the organization is registered.

Do not abbreviate the organization name and do not use the following characters in the name:

Angle brackets (< >) tilde (~), exclamation mark, at (@), pound (#), zero (0), caret (^), asterisk (*), forward slash (/), square brackets ([ ]), question mark (?).

organizationUnitName

Name of the division or section in the organization that will use the certificate.

localityName

Name of the city or town in which your organization's head office is located.

commonName

Fully qualified domain name for the company or web site. The common name must match the name used by DNS servers to do a DNS lookup of your server. Most browsers use this information for authenticating the server's certificate during the SSL handshake. If the server name in the URL does not match the common name as given in the server certificate, the browser terminates the SSL handshake or prompts the user with a warning message.

Do not use wildcard characters, such as asterisk (*) or question mark (?), and do not use an IP address as the common name. The common name must not contain the protocol specifier <http://> or <https://>.

emailAddress

Contact person's e-mail address. This address is publically displayed as part of the certificate. Provide an e-mail address that is monitored by an administrator who can be contacted about the certificate.

challengePassword

Pass phrase, embedded in the certificate signing request that is shared only between the client or server requesting the certificate and the SSL certificate issuer (typically the certificate authority). This pass phrase can be used to authenticate a client or server that is requesting a certificate from the certificate authority.

companyName

Additional name for the company or web site.

Example

```
create ssl certreq /nsconfig/ssl/csr.pem -keyFile /nsconfig/ssl/rsa1024.pem
```

ssl cipher

Sep 22, 2015

The following operations can be performed on "ssl cipher":

add | bind | show | rm | unbind

add ssl cipher

Creates a user-defined cipher group, which you can bind to an SSL virtual server instead of binding ciphers individually. Although you cannot modify a built-in cipher group, you can add built-in cipher groups as well as individual ciphers to a user-defined cipher group.

Synopsys

add ssl cipher <cipherGroupName>

Arguments

cipherGroupName

Name for the user-defined cipher group. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the cipher group is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my ciphergroup" or 'my ciphergroup').

cipherAliasName/cipherName/cipherGroupName

The individual cipher name(s), a user-defined cipher group, or a system predefined cipher alias that will be added to the predefined cipher alias that will be added to the group cipherGroupName.

If a cipher alias or a cipher group is specified, all the individual ciphers in the cipher alias or group will be added to the user-defined cipher group.

Example

1) add ssl cipher mygroup SSL2-RC4-MD5 SSL2-EXP-RC4-MD5 The above command creates a new cipher-group by the name: mygroup, with the two ciphers SSL2-RC4-MD5

---

bind ssl cipher

Adds ciphers to a user-defined cipher group. You can add an existing cipher group to a user-defined cipher group but you cannot modify a built-in cipher group.

Synopsys

bind ssl cipher [<cipherGroupName>@] [-cipherName <string>]

Arguments

cipherGroupName

Name of the user-defined cipher group.

vServerName

The name of the SSL virtual server to which the cipher-suite is to be bound.

serviceName

The name of the SSL service name to which the cipher-suite is to be bound.

serviceGroupName

The name of the SSL service name to which the cipher-suite is to be bound.

cipherOperation

The operation that is performed when adding the cipher-suite.

Possible cipher operations are:

ADD - Appends the given cipher-suite to the existing one configured for the virtual server.

REM - Removes the given cipher-suite from the existing one configured for the virtual server.

ORD - Overrides the current configured cipher-suite for the virtual server with the given cipher-suite.

Possible values: ADD, REM, ORD

cipherAliasName/cipherName/cipherGroupName

A cipher-suite can consist of an individual cipher name, the system predefined cipher-alias name, or user defined cipher-group name.

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias to add to the cipher group.

Example

1) bind ssl cipher sslvip ADD SSL3-RC4-SHA The above example appends the cipher SSL3-RC4-SHA to the cipher-suite already configured for the SSL virtual server sslvip.

---

show ssl cipher

Displays information about all the cipher groups defined on the appliance, or displays detailed information about the specified cipher group.

Synopsys

show ssl cipher [<cipherGroupName>]

Arguments

cipherGroupName

Name of the cipher group for which to show detailed information.

summary

fullValues

format

level

Outputs

description

Cipher suite description.

cipherName

Cipher name.

flag

stateflag

peFlags

devno

count

Example

1) An example of the output of the show ssl cipher SSL3-RC4-MD5 command is as follows: Cipher Name: SSL3-RC4-MD5

---

rm ssl cipher

Removes a user-defined cipher group from the appliance.

Synopsys

rm ssl cipher <cipherGroupName>

Arguments

cipherGroupName

Name of the user-defined cipher group to remove.

cipherName

The cipher(s) to be removed from the cipher group.

Example

1) rm ssl cipher mygroup SSL2-RC4-MD5 The above example removes the cipher SSL2-RC4-MD5 from the cipher group m

---

unbind ssl cipher

Removes all the ciphers from a user-defined cipher group. You can only remove individual ciphers from a user-defined cipher group. Removing groups is not supported.

Synopsys

unbind ssl cipher <cipherGroupName> [-cipherName <string> ...]

Arguments

cipherGroupName

Name of the user-defined cipher group.

cipherName

Name(s) of the cipher(s) to be removed from the user-defined cipher group.

Example

1) rm ssl cipher mygroup SSL2-RC4-MD5 The above example removes the cipher SSL2-RC4-MD5 from the cipher group m

ssl ciphersuite

Sep 22, 2015

The following operations can be performed on "ssl ciphersuite":

show ssl ciphersuite

Displays information about all the cipher suites configured on the appliance, or displays detailed information about the specified cipher-suite. A cipher suite comprises a protocol and the following algorithms: key exchange (Kx), authentication (Au), encryption (Enc), and message authentication code (Mac).

Synopsys

show ssl ciphersuite [<cipherName>]

Arguments

cipherName

Name of the cipher suite for which to show detailed information.

summary

fullValues

format

level

Outputs

description

Cipher suite description.

flag

stateflag

devno

count

Example

1) An example of the output of the show ssl cipher SSL3-RC4-MD5 command is as follows: Cipher Name: SSL3-RC4-MD5 Descri

ssl crl

Sep 22, 2015

The following operations can be performed on "ssl crl":

add | create | rm | set | unset | show

add ssl crl

Adds a Certificate Revocation List (CRL). A CRL identifies invalid certificates by serial number and issuer. In a high availability configuration, the CRL must be in the same location on the primary and secondary nodes.

Synopsys

add ssl crl <crlName> <crlPath> [-inform (DER | PEM)] [-refresh (ENABLED | DISABLED)] [-CAcert <string>] [-method (HTTP | LDAP)] [-server <ip_addr|ipv6_addr|*> | -url <URL>] [-port <port>] [-baseDN <string>] [-scope (Base | One)] [-interval <interval>] [-day <integer>] [-time <HH:MM>] [-bindDN <string>] {-password } [-binary (YES | NO)]

Arguments

crlName

Name for the Certificate Revocation List (CRL). Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the CRL is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my crl" or 'my crl').

crlPath

Path to the CRL file. /var/netscaler/ssl/ is the default path.

inform

Input format of the CRL file. The two formats supported on the appliance are:

PEM - Privacy Enhanced Mail.

DER - Distinguished Encoding Rule.

Possible values: DER, PEM

Default value: FORMAT_PEM

refresh

Set CRL auto refresh.

Possible values: ENABLED, DISABLED

CAcert

CA certificate that has issued the CRL. Required if CRL Auto Refresh is selected. Install the CA certificate on the appliance before adding the CRL.

method

Method for CRL refresh. If LDAP is selected, specify the method, CA certificate, base DN, port, and LDAP server name. If HTTP is selected, specify the CA certificate, method, URL, and port. Cannot be changed after a CRL is added.

Possible values: HTTP, LDAP

server

IP address of the LDAP server from which to fetch the CRLs.

url

URL of the CRL distribution point.

port

Port for the LDAP server.

Minimum value: 1

baseDN

Base distinguished name (DN), which is used in an LDAP search to search for a CRL. Citrix recommends searching for the Base DN instead of the Issuer Name from the CA certificate, because the Issuer Name field might not exactly match the LDAP directory structure's DN.

scope

Extent of the search operation on the LDAP server. Available settings function as follows:

One - One level below Base DN.

Base - Exactly the same level as Base DN.

Possible values: Base, One

Default value: NSAPI_ONESCOPE

interval

CRL refresh interval. Use the NONE setting to unset this parameter.

Possible values: MONTHLY, WEEKLY, DAILY, NONE

day

Day on which to refresh the CRL, or, if the Interval parameter is not set, the number of days after which to refresh the CRL. If Interval is set to MONTHLY, specify the date. If Interval is set to WEEKLY, specify the day of the week (for example, Sun=0 and Sat=6). This parameter is not applicable if the Interval is set to DAILY.

Maximum value: 31

time

Time, in hours (1-24) and minutes (1-60), at which to refresh the CRL.

bindDN

Bind distinguished name (DN) to be used to access the CRL object in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

password

Password to access the CRL in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

binary

Set the LDAP-based CRL retrieval mode to binary.

Possible values: YES, NO

Default value: NO

Example

1) add ssl certkey CAcert -cert /nsconfig/ssl/ca_cert.pem add ssl crl crl_file /var/netscaler/ssl/crl.pem -cacert CAcert The above command adds a CRL from local storage sys

---

create ssl crl

Revokes a certificate, or list of certificates, or generates a CRL for the list of revoked certificates.

Synopsys

create ssl crl <CAcertFile> <CAkeyFile> <indexFile> (-revoke <input_filename> | -genCRL <output_filename>) {-password }

Arguments

CAcertFile

Name of and, optionally, path to the CA certificate file.

/nsconfig/ssl/ is the default path.

Maximum value: 63

CAkeyFile

Name of and, optionally, path to the CA key file. /nsconfig/ssl/ is the default path

Maximum value: 63

indexFile

Name of and, optionally, path to the file containing the serial numbers of all the certificates that are revoked. Revoked certificates are appended to the file. /nsconfig/ssl/ is the default path

Maximum value: 63

revoke

Name of and, optionally, path to the certificate to be revoked. /nsconfig/ssl/ is the default path.

Maximum value: 63

genCRL

Name of and, optionally, path to the CRL file to be generated. The list of certificates that have been revoked is obtained from the index file. /nsconfig/ssl/ is the default path.

Maximum value: 63

password

Password for the CA key file.

Maximum value: 31

Example

1) create crl /nsconfig/ssl/cacert.pem /nsconfig/ssl/cakey.pem /nsconfig/ssl/index.txt -gencrl /var/netscaler/ssl/crl.pem

---

rm ssl crl

Removes the specified CRL from the appliance.

Synopsys

rm ssl crl <crlName> ...

Arguments

crlName

Name of the CRL to remove.

Example

1) rm ssl crl ca_crl The above CLI command to delete the CRL object ca_crl from the system is.

---

set ssl crl

Modifies all the parameters of a CRL, except the CRL name and method.

Synopsys

set ssl crl <crlName> [-refresh (ENABLED | DISABLED)] [-CAcert <string>] [-server <ip_addr|ipv6_addr|*> | -url <URL>] [-method (HTTP | LDAP)] [-port <port>] [-baseDN <string>] [-scope (Base | One)] [-interval <interval>] [-day <integer>] [-time <HH:MM>] [-bindDN <string>] { -password } [-binary (YES | NO)]

Arguments

crlName

Name of the CRL to be modified.

refresh

Set CRL auto refresh.

Possible values: ENABLED, DISABLED

CAcert

CA certificate that has issued the CRL. Required if CRL Auto Refresh is selected. Install the CA certificate on the appliance before adding the CRL.

server

IP address of the LDAP server from which to fetch the CRLs.

method

Method for CRL refresh. If LDAP is selected, specify the method, CA certificate, base DN, port, and LDAP server name. If HTTP is selected, specify the CA certificate, method, URL, and port. Cannot be changed after a CRL is added.

Possible values: HTTP, LDAP

port

Port for the LDAP server.

Minimum value: 1

baseDN

Base distinguished name (DN), which is used in an LDAP search to search for a CRL. Citrix recommends searching for the Base DN instead of the Issuer Name from the CA certificate, because the Issuer Name field might not exactly match the LDAP directory structure's DN.

scope

Extent of the search operation on the LDAP server. Available settings function as follows:

One - One level below Base DN.

Base - Exactly the same level as Base DN.

Possible values: Base, One

Default value: NSAPI_ONESCOPE

interval

CRL refresh interval. Use the NONE setting to unset this parameter.

Possible values: MONTHLY, WEEKLY, DAILY, NOW, NONE

day

Day on which to refresh the CRL, or, if the Interval parameter is not set, the number of days after which to refresh the CRL. If Interval is set to MONTHLY, specify the date. If Interval is set to WEEKLY, specify the day of the week (for example, Sun=0 and Sat=6). This parameter is not applicable if the Interval is set to DAILY.

Maximum value: 31

time

Time, in hours (1-24) and minutes (1-60), at which to refresh the CRL.

bindDN

Bind distinguished name (DN) to be used to access the CRL object in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

password

Password to access the CRL in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

binary

Set the LDAP-based CRL retrieval mode to binary.

Possible values: YES, NO

Default value: NO

Example

1) set ssl crl crl_file -refresh ENABLE -interval MONTHLY -days 10 -time 12:00 The above example sets the CRL refresh to every Month, on date=10, and time=12:00hrs. 2

---

unset ssl crl

Use this command to remove ssl crl settings.Refer to the set ssl crl command for meanings of the arguments.

Synopsys

unset ssl crl <crlName> [-refresh] [-CAcert] [-server] [-method] [-url] [-port] [-baseDN] [-scope] [-interval] [-day] [-time] [-bindDN] [-password] [-binary]

---

show ssl crl

Displays information about all the CRLs configured on the appliance, or displays detailed information about the specified CRL.

Synopsys

show ssl crl [<crlName>]

Arguments

crlName

Name of the CRL for which to show detailed information.

summary

fullValues

format

level

Outputs

crlPath

The name and path to the file containing the CRL.

inform

The encoding format of the CRL (PEM or DER).

CAcert

The CA certificate that issued the CRL.

refresh

The state of the auto refresh feature for the CRL.

scope

Extent of the search operation on the LDAP server.

Base: Exactly the same level as basedn

One : One level below basedn.

server

The IP address of the LDAP/HTTP server from which the CRLs are to be fetched.

port

The port of the LDAP/HTTP server.

url

URL of the CRL distribution point.

method

The method for CRL refresh (LDAP or HTTP).

baseDN

The baseDN to be used to fetch the CRL object from the LDAP server.

interval

The CRL refresh interval.

day

The day when the CRL is to be refreshed.

time

The time when the CRL is to be refreshed.

bindDN

The bindDN to be used to access the CRL object in the LDAP repository.

password

The password to be is used to access the CRL object in the LDAP repository.

flags

CRL status flag.

lastupdatetime

Last CRL refresh time.

version

CRL version.

signaturealgo

Signature algorithm.

issuer

Issuer name.

lastupdate

Last update time.

nextupdate

Next update time.

date

Certificate Revocation date

number

Certificate Serial number.

binary

Mode of retrieval of CRL from LDAP server.

daysToExpiration

Number of days remaining for the CRL to expire.

devno

count

stateflag

Example

1) An example output of the show ssl crl command is as follows: 1 configured CRL(s) 1 Name: ca_crl CRL Path: /var/netscaler/ssl/cr1.d

ssl dhParam

Sep 22, 2015

The following operations can be performed on "ssl dhParam":

create ssl dhParam

Generates a Diffie-Hellman (DH) key.

Synopsys

create ssl dhParam <dhFile> [<bits>] [-gen (2 | 5)]

Arguments

dhFile

Name of and, optionally, path to the DH key file. /nsconfig/ssl/ is the default path.

Maximum value: 63

bits

Size, in bits, of the DH key being generated.

Minimum value: 512

Maximum value: 2048

gen

Random number required for generating the DH key. Required as part of the DH key generation algorithm.

Possible values: 2, 5

Default value: 2

Example

1) create ssl dhparam /nsconfig/ssl/dh1024.pem 1024 -gen 5

ssl dsaKey

Sep 22, 2015

The following operations can be performed on "ssl dsaKey":

create ssl dsaKey

Generates a DSA key.

Synopsys

create ssl dsaKey <keyFile> <bits> [-keyform (DER | PEM)] [-des | -des3] {-password }

Arguments

keyFile

Name for and, optionally, path to the DSA key file. /nsconfig/ssl/ is the default path.

Maximum value: 63

bits

Size, in bits, of the DSA key.

Minimum value: 512

Maximum value: 2048

keyform

Format in which the DSA key file is stored on the appliance.

Possible values: DER, PEM

Default value: FORMAT_PEM

des

Encrypt the generated DSA key by using the DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that will be used to encrypt the key.

des3

Encrypt the generated DSA key by using the Triple-DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that will be used to encrypt the key.

password

Pass phrase to use for encryption if DES or DES3 option is selected.

Maximum value: 31

Example

```
create ssl dsakey /nsconfig/ssl/dsa1024.pem 1024
```

ssl dtlsProfile

Sep 22, 2015

The following operations can be performed on "ssl dtlsProfile":

add | rm | set | unset | show

add ssl dtlsProfile

Add a new DTLS profile on the Netscaler

Synopsys

add ssl dtlsProfile <name> [-pmtuDiscovery (ENABLED | DISABLED)] [-maxRecordSize <positive_integer>] [-maxRetryTime <positive_integer>] [-helloVerifyRequest (ENABLED | DISABLED)] [-terminateSession (ENABLED | DISABLED)] [-maxPacketSize <positive_integer>]

Arguments

name

Name of the DTLS profile

pmtuDiscovery

Enable/disable PMTU discovery

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxRecordSize

Maximum record size

Default value: 1460

Maximum value: 1472

maxRetryTime

Maximum retry time

Default value: 3

helloVerifyRequest

Enable/disable sending of hello verify request

Possible values: ENABLED, DISABLED

Default value: DISABLED

terminateSession

Option to decide whether to terminate the session if bad MAC decode error happens.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxPacketSize

Max packet count to reassemble for a fragmented record ? attack protection if client send small packets

Default value: 120

Maximum value: 86400

Example

```
add dtlsProfile dtls1 -helloVerifyRequest  ENABLED -maxRetryTime 4
```

---

rm ssl dtlsProfile

Remove a DTLS profile on the Netscaler

Synopsys

```
rm ssl dtlsProfile <name>
```

Arguments

name

Name of the DTLS profile

Example

```
rm dtlsprofile <profile name>
```

---

set ssl dtlsProfile

Set/modify DTLS profile values

Synopsys

```
set ssl dtlsProfile <name> [-pmtuDiscovery ( ENABLED | DISABLED )] [-maxRecordSize <positive_integer>] [-maxRetryTime <positive_integer>] [-helloVerifyRequest ( ENABLED | DISABLED )] [-terminateSession ( ENABLED | DISABLED )] [-maxPacketSize <positive_integer>]
```

Arguments

name

Name of the DTLS profile

pmtuDiscovery

Enable/disable PMTU discovery

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxRecordSize

Maximum record size

Default value: 1460

Maximum value: 1472

maxRetryTime

Maximum retry time

Default value: 3

helloVerifyRequest

Enable/disable sending of hello verify request

Possible values: ENABLED, DISABLED

Default value: DISABLED

terminateSession

Option to decide whether to terminate the session if bad MAC decode error happens.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxPacketSize

Max packet count to reassemble for a fragmented record ? attack protection if client send small packets

Default value: 120

Maximum value: 86400

Example

```
set dtlsprofile <profile name> -dropInvalReqs ON -markHttp09Inval ON
```

---

unset ssl dtlsProfile

Use this command to remove ssl dtlsProfile settings.Refer to the set ssl dtlsProfile command for meanings of the arguments.

Synopsys

unset ssl dtlsProfile <name> [-pmtuDiscovery] [-maxRecordSize] [-maxRetryTime] [-helloVerifyRequest] [-terminateSession] [-maxPacketSize]

---

show ssl dtlsProfile

Display all the configured DTLS profiles in the system. If a name is specified, then only that profile is shown.

Synopsys

show ssl dtlsProfile [<name>]

Arguments

name

Name of the DTLS profile.

summary

fullValues

format

level

Outputs

pmtuDiscovery

PMTU Discovery

maxRecordSize

Maximum record size

maxRetryTime

Maximum retry time

helloVerifyRequest

Hello Verify Request

terminateSession

Terminate Session

maxPacketSize

Maximum Packet Size

devno

count

stateflag

Example

show dtls profile [profile name]

ssl fips

Sep 22, 2015

The following operations can be performed on "ssl fips":

set | unset | reset | show | update

set ssl fips

Initializes the Hardware Security Module (HSM) on the FIPS card and sets a new security officer password and user password. CAUTION: This command erases all data on the FIPS card. You are prompted before proceeding with the command execution. A restart is required before and after executing this command for the changes to apply. Save the configuration after executing this command and before restarting the appliance.

Synopsys

set ssl fips -initHSM Level-2 [-hsmLabel <string>]

Arguments

initHSM

FIPS initialization level. The appliance currently supports Level-2 (FIPS 140-2).

Possible values: Level-2

soPassword

Security officer password that will be in effect after you have configured the HSM.

oldSoPassword

Old password for the security officer.

userPassword

The Hardware Security Module's (HSM) User password.

hsmLabel

Label to identify the Hardware Security Module (HSM).

Example

1) set fips -initHSM Level-2 fipsso123 oldfipsso123 fipuser123 -hsmLabel FIPS-140-2 >This command will erase all data on the FIPS card. You must save the configuratio

---

unset ssl fips

Use this command to remove ssl fips settings.Refer to the set ssl fips command for meanings of the arguments.

Synopsys

unset ssl fips -hsmLabel

---

reset ssl fips

Resets the FIPS card to the default password for Security Officer and User accounts. This command can be used only if the FIPS card has been locked because of three or more unsuccessful login attempts.

Synopsys

reset ssl fips

Example

reset fips

---

show ssl fips

Displays the information on the FIPS card.

Synopsys

show ssl fips

Arguments

format

level

Outputs

initHSM

The level of the FIPS initialization.

soPassword

Security officer password that will be in effect after you have configured the HSM.

userPassword

The Hardware Security Module's (HSM) User password.

oldSoPassword

Old password for the security officer.

eraseData

Erase data.

hsmLabel

FIPS card (HSM) label

serial

FIPS card serial number.

majorVersion

Firmware major version.

minorVersion

Firmware minor version.

FipsHwMajorVersion

FIPS card hardware major version.

FipsHwMinorVersion

FIPS card hardware minor version.

FipsHwVersionString

FIPS card hardware extended version string.

flashMemoryTotal

Total size of the flash memory on card.

flashMemoryFree

Total size of free flash memory.

sramTotal

Total size of the SRAM memory on card.

sramFree

Total size of free SRAM memory.

status

Status.

flag

Internal Flags.

serialNo

FIPS card serial number.

state

FIPS card state.

firmwareReleaseDate

FIPS card firmware revision date.

coresMax

Maximum number of crypto cores present in the FIPS card.

coresEnabled

Number of crypto cores enabled in the FIPS card.

Example

An example of the output for show ssl fips command is as follows: FIPS HSM Info: HSM Label : FIPS1 Initialization : FIPS-140-2 Level-2 HSM Seria

---

update ssl fips

Updates the FIPS firmware. Note: Only compatible firmware version upgrade is allowed. For example, 4.6.0 to 4.6.1

Synopsys

update ssl fips -fipsFW 4.6.1

Arguments

fipsFW

FIPS firmware update.

Possible values: 4.6.1

Example

update ssl fips -fipsFW 4.6.1

---

ssl fipsKey

Sep 22, 2015

The following operations can be performed on "ssl fipsKey":

create | rm | show | import | export

create ssl fipsKey

Generates a FIPS key within the Hardware Security Module (HSM) of the FIPS card.

Synopsys

create ssl fipsKey <fipsKeyName> -modulus <positive_integer> [-exponent (3 | F4)]

Arguments

fipsKeyName

Name for the FIPS key. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the FIPS key is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my fipskey" or 'my fipskey').

modulus

Modulus, in multiples of 64, of the FIPS key to be created.

Minimum value: 1024

Maximum value: 4096

exponent

Exponent value for the FIPS key to be created. Available values function as follows:

3=3 (hexadecimal)

F4=10001 (hexadecimal)

Possible values: 3, F4

Default value: 3

Example

```
create fipskey fips1 -modulus 1024 -exp f4
```

---

rm ssl fipsKey

Removes all the FIPS keys, or the specified FIPS key, from the appliance.

Synopsys

rm ssl fipsKey <fipsKeyName> ...

Arguments

fipsKeyName

Name of the FIPS key to remove.

Example

```
rm fipskey fips1
```

---

show ssl fipsKey

Displays information about all the FIPS keys configured on the appliance, or displays detailed information about the specified FIPS key.

Synopsys

show ssl fipsKey [<fipsKeyName>]

Arguments

fipsKeyName

Name of the FIPS key for which to show detailed information.

summary

fullValues

format

level

Outputs

modulus

The modulus of the key.

exponent

The exponent value for the key.

size

Size.

devno

count

stateflag

Example

1) An example of output of show ssl fipskey command is as follows: show fipskey 2 FIPS keys: 1) FIPS Key Name: fips1 2) FIPS K

---

import ssl fipsKey

Imports a FIPS key into the Hardware Security Module (HSM) of the FIPS card. Can import an existing FIPS key, or can import, as a FIPS key, an external private key, such as a key that was created on an Apache or IIS external Web server.

Synopsys

import ssl fipsKey <fipsKeyName> -key <string> [-inform <inform>] [-wrapKeyName <string>] [-iv <string>] [-exponent (3 | F4)]

Arguments

fipsKeyName

Name for the FIPS key to be imported. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the FIPS key is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my fipskey" or 'my fipskey').

key

Name of and, optionally, path to the key file to be imported.

/nsconfig/ssl/ is the default path.

inform

Input format of the key file. Available formats are:

SIM - Secure Information Management; select when importing a FIPS key. If the external FIPS key is encrypted, first decrypt it, and then import it.

PEM - Privacy Enhanced Mail; select when importing a non-FIPS key.

Possible values: SIM, DER, PEM

Default value: FORMAT_SIM

wrapKeyName

Name of the wrap key to use for importing the key. Required for importing a non-FIPS key.

iv

Initialization Vector (IV) to use for importing the key. Required for importing a non-FIPS key.

exponent

Exponent value for the FIPS key to be created. Available values function as follows:

3=3 (hexadecimal)

F4=10001 (hexadecimal)

Possible values: 3, F4

Default value: 3

Example

1) import fipskey fips1 -key /nsconfig/ssl/fipskey.sim The above example imports a FIPS key stored in the file fipskey.sim in the systen

---

export ssl fipsKey

Exports a FIPS key from one appliance to another or backs up a FIPS key in a secure manner. The exported key is secured by using a strong asymmetric key encryption method.

Synopsys

export ssl fipsKey <fipsKeyName> -key <string>

Arguments

fipsKeyName

Name of the FIPS key to export.

key

Name of and, optionally, path to the exported key file.

/nsconfig/ssl/ is the default path.

Example

export fipskey fips1 -key /nsconfig/ssl/fips1.key

---

ssl fipsSIMSource

Sep 22, 2015

The following operations can be performed on "ssl fipsSIMSource":

enable | init

enable ssl fipsSIMSource

Enable the source FIPS appliance to participate in a secure exchange of keys with the target (secondary) FIPS appliance.

Synopsys

enable ssl fipsSIMSource <targetSecret> <sourceSecret>

Arguments

targetSecret

Name of and, optionally, path to the target FIPS appliance's secret data. /nsconfig/ssl/ is the default path.

sourceSecret

Name for and, optionally, path to the source FIPS appliance's secret data. /nsconfig/ssl/ is the default path.

Example

```
enable fipsSIMsource /nsconfig/ssl/target.secret /nsconfig/ssl/source.secret
```

---

init ssl fipsSIMSource

Initialize the source FIPS appliance for participating in a secure exchange of keys with the target (secondary) FIPS appliance.

Synopsys

init ssl fipsSIMSource <certFile>

Arguments

certFile

Name for and, optionally, path to the source FIPS appliance's certificate file. /nsconfig/ssl/ is the default path.

Example

```
init fipsSIMsource /nsconfig/ssl/source.cert
```

---

ssl fipsSIMTarget

Sep 22, 2015

The following operations can be performed on "ssl fipsSIMTarget":

enable | init

enable ssl fipsSIMTarget

Enables secure transfer of FIPS keys in a high availability setup from the primary appliance to the secondary appliance.

Synopsys

enable ssl fipsSIMTarget <keyVector> <sourceSecret>

Arguments

keyVector

Name of and, optionally, path to the target FIPS appliance's key vector. /nsconfig/ssl/ is the default path.

sourceSecret

Name of and, optionally, path to the source FIPS appliance's secret data. /nsconfig/ssl/ is the default path.

Example

```
enable fipsSIMtarget /nsconfig/ssl/target.key /nsconfig/ssl/source.secret
```

---

init ssl fipsSIMTarget

Initialize the target (secondary) FIPS appliance for participating in a secure exchange of keys with the primary FIPS appliance.

Synopsys

init ssl fipsSIMTarget <certFile> <keyVector> <targetSecret>

Arguments

certFile

Name of and, optionally, path to the source FIPS appliance's certificate file. /nsconfig/ssl/ is the default path.

keyVector

Name for and, optionally, path to the target FIPS appliance's key vector. /nsconfig/ssl/ is the default path.

targetSecret

Name for and, optionally, path to the target FIPS appliance's secret data. The default input path for the secret data is /nsconfig/ssl/.

Example

```
init fipsSIMtarget /nsconfig/ssl/source.cert /nsconfig/ssl/target.key /nsconfig/ssl/target.secret
```

ssl global

Sep 22, 2015

The following operations can be performed on "ssl global":

bind | unbind | show

bind ssl global

Binds an SSL policy globally.

Synopsys

bind ssl global [-policyName <string>] [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type <type>] [-invoke (<labelType> <labelName>)]

Arguments

policyName

Name of the SSL policy.

Example

```
bind ssl global -policyName certInsert_pol -priority 100
```

---

unbind ssl global

Unbinds a globally bound SSL policy.

Synopsys

unbind ssl global [-policyName <string> [-type <type>] [-priority <positive_integer>]]

Arguments

policyName

Name of the SSL policy to unbind.

Example

```
unbind ssl global -policyName certInsert_pol
```

---

show ssl global

Displays globally bound SSL policies.

Synopsys

show ssl global [-type <type>]

Arguments

type

Global bind point to which the policy is bound.

Possible values: CONTROL_OVERRIDE, CONTROL_DEFAULT, DATA_OVERRIDE, DATA_DEFAULT

summary

fullValues

format

level

Outputs

stateflag

policyName

The name for the SSL policy.

priority

The priority of the policy binding.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

numpol

The number of policies bound to the bindpoint.

devno

count

Example

show ssl global 1 Globally Active SSL Policy: 1) Name: certInsert_pol Priority: 100

ssl ocspResponder

Sep 22, 2015

The following operations can be performed on "ssl ocspResponder":

add | rm | set | unset | show

add ssl ocspResponder

Adds an OCSP responder. An OCSP responder identifies the OCSP server that validates a certificate. NetScaler appliances support OCSP as defined in RFC 2560.

Synopsys

add ssl ocspResponder <name> -url <URL> [-cache (ENABLED | DISABLED) [-cacheTimeout <positive_integer>]] [-batchingDepth <positive_integer>] [-batchingDelay <positive_integer>] [-resptimeout <positive_integer>] [-responderCert <string> | -trustResponder] [-producedAtTimeSkew <positive_integer>] [-signingCert <string>] [-useNonce (YES | NO)] [-insertClientCert (YES | NO)]

Arguments

name

Name for the OCSP responder. Cannot begin with a hash (#) or space character and must contain only ASCII alphanumeric, underscore (_), hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the responder is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder" or 'my responder').

url

URL of the OCSP responder.

cache

Enable caching of responses. Caching of responses received from the OCSP responder enables faster responses to the clients and reduces the load on the OCSP responder.

Possible values: ENABLED, DISABLED

cacheTimeout

Timeout for caching the OCSP response. After the timeout, the NetScaler sends a fresh request to the OCSP responder for the certificate status. If a timeout is not specified, the timeout provided in the OCSP response applies.

Default value: 1

Minimum value: 1

Maximum value: 1440

batchingDepth

Number of client certificates to batch together into one OCSP request. Batching avoids overloading the OCSP responder. A value of 1 signifies that each request is queried independently. For a value greater than 1, specify a timeout (batching delay) to avoid inordinately delaying the processing of a single certificate.

Minimum value: 1

Maximum value: 8

batchingDelay

Maximum time, in milliseconds, to wait to accumulate OCSP requests to batch. Does not apply if the Batching Depth is 1.

Maximum value: 10000

resptimeout

Time, in milliseconds, to wait for an OCSP response. When this time elapses, an error message appears or the transaction is forwarded, depending on the settings on the virtual server. Includes Batching Delay time.

Maximum value: 120000

responderCert

producedAtTimeSkew

Time, in seconds, for which the NetScaler waits before considering the response as invalid. The response is considered invalid if the Produced At time stamp in the OCSP response exceeds or precedes the current NetScaler clock time by the amount of time specified.

Default value: 300

Maximum value: 86400

signingCert

Certificate-key pair that is used to sign OCSP requests. If this parameter is not set, the requests are not signed.

useNonce

Enable the OCSP nonce extension, which is designed to prevent replay attacks.

Possible values: YES, NO

insertClientCert

Include the complete client certificate in the OCSP request.

Possible values: YES, NO

Example

1) add ssl ocspResponder -url http://ocsp.example.com -producedAtTimeSkew 0 The above command will only allow responses that were generated in the same second to l

---

rm ssl ocspResponder

Removes the specified OCSP responder from the appliance.

Synopsys

rm ssl ocspResponder <name> ...

Arguments

name

Name of the OCSP responder to remove. The OCSP responder is removed only if it is not referenced by any other object.

Example

1) rm ssl ocspResponder o1 The above command removes the OCSP responder o1 from the system.

---

set ssl ocspResponder

Modifies the parameters of an OCSP responder.

Synopsys

set ssl ocspResponder <name> [-url <URL>] [-cache (ENABLED | DISABLED)] [-cacheTimeout <positive_integer>] [-batchingDepth <positive_integer>] [-batchingDelay <positive_integer>] [-resptimeout <positive_integer>] [-responderCert <string> | -trustResponder] [-producedAtTimeSkew <positive_integer>] [-signingCert <string>] [-useNonce (YES | NO)] [-insertClientCert (YES | NO)]

Arguments

name

Name of the OCSP responder to modify.

url

URL of the OCSP responder.

cache

Enable caching of responses. Caching of responses received from the OCSP responder enables faster responses to the clients and reduces the load on the OCSP responder.

Possible values: ENABLED, DISABLED

cacheTimeout

Timeout for caching the OCSP response. After the timeout, the NetScaler sends a fresh request to the OCSP responder for the certificate status. If a timeout is not specified, the timeout provided in the OCSP response applies.

Default value: 1

Minimum value: 1

Maximum value: 1440

batchingDepth

Number of client certificates to batch together into one OCSP request. Batching avoids overloading the OCSP responder. A value of 1 signifies that each request is queried independently. For a value greater than 1, specify a timeout (batching delay) to avoid inordinately delaying the processing of a single certificate.

Minimum value: 1

Maximum value: 8

batchingDelay

Maximum time, in milliseconds, to wait to accumulate OCSP requests to batch. Does not apply if the Batching Depth is 1.

Maximum value: 10000

resptimeout

Time, in milliseconds, to wait for an OCSP response. When this time elapses, an error message appears or the transaction is forwarded, depending on the settings on the virtual server. Includes Batching Delay time.

Maximum value: 120000

responderCert

producedAtTimeSkew

Time, in seconds, for which the NetScaler waits before considering the response as invalid. The response is considered invalid if the Produced At time stamp in the OCSP response exceeds or precedes the current NetScaler clock time by the amount of time specified.

Default value: 300

Maximum value: 86400

signingCert

Certificate-key pair that is used to sign OCSP requests. If this parameter is not set, the requests are not signed.

useNonce

Enable the OCSP nonce extension, which is designed to prevent replay attacks.

Possible values: YES, NO

insertClientCert

Include the complete client certificate in the OCSP request.

Possible values: YES, NO

Example

1) add ssl ocspResponder -url http://ocsp.example.com -producedAtTimeSkew 0 The above command will only allow responses that were generated in the same second to l

---

unset ssl ocspResponder

Removes the attributes of an OCSP responder. Attributes for which a default value is available revert to their default values. Refer to the set ssl ocspResponder command for descriptions of the arguments..Refer to the set ssl ocspResponder command for meanings of the arguments.

Synopsys

unset ssl ocspResponder <name> [-trustResponder] [-insertClientCert (YES | NO)] [-cache] [-cacheTimeout] [-batchingDepth] [-batchingDelay] [-resptimeout] [-responderCert] [-producedAtTimeSkew] [-signingCert] [-useNonce]

---

show ssl ocspResponder

Displays information about all the OCSP responders configured on the appliance, or displays detailed information about the specified OCSP responder.

Synopsys

show ssl ocspResponder [<name>]

Arguments

name

Name of the OCSP responder for which to show detailed information.

summary

fullValues

format

level

Outputs

url

URL of the OCSP responder.

useAIA

Only use the URL present in the certificate.

cache

Enable caching of responses. Caching of responses received from the OCSP responder enables faster responses to the clients and reduces the load on the OCSP responder.

cacheTimeout

Timeout for caching the OCSP response. After the timeout, the NetScaler sends a fresh request to the OCSP responder for the certificate status. If a timeout is not specified, the timeout provided in the OCSP response applies.

batchingDepth

Number of client certificates to batch together into one OCSP request. Batching avoids overloading the OCSP responder. A value of 1 signifies that each request is queried independently. For a value greater than 1, specify a timeout (batching delay) to avoid inordinately delaying the processing of a single certificate.

batchingDelay

Maximum time, in milliseconds, to wait to accumulate OCSP requests to batch. Does not apply if the Batching Depth is 1.

resptimeout

Maximum time, in mS, to wait for an OCSP response before giving up. Defaults to 2000 mS. If this is set to 0, NetScaler will wait for an indefinite amount of time.

producedAtTimeSkew

Time, in seconds, for which the NetScaler waits before considering the response as invalid. The response is considered invalid if the Produced At time stamp in the OCSP response exceeds or precedes the current NetScaler clock time by the amount of time specified.

responderCert

trustResponder

A certificate to use to validate OCSP responses. Alternatively, if -trustResponder is specified, no verification will be done on the reponse. If both are omitted, only the response times (producedAt, lastUpdate, nextUpdate) will be verified.

signingCert

Certificate-key pair that is used to sign OCSP requests. If this parameter is not set, the requests are not signed.

useNonce

Add a nonce to the OCSP request. Protects against replay attacks.

dns

Was DNS resolution successful for a domain-based OCSP responder

insertClientCert

Include the complete client certificate in the OCSP request.

IPAddress

The IPv6 address of the ocsp responder.

devno

count

stateflag

ssl parameter

Sep 22, 2015

The following operations can be performed on "ssl parameter":

set | unset | show

set ssl parameter

Synopsys

set ssl parameter [-quantumSize <quantumSize>] [-crlMemorySizeMB <positive_integer>] [-strictCAChecks (YES | NO)] [-sslTriggerTimeout <positive_integer>] [-sendCloseNotify (YES | NO)] [-encryptTriggerPktCount <positive_integer>] [-denySSLReneg <denySSLReneg>] [-insertionEncoding (Unicode | UTF-8)] [-ocspCacheSize <positive_integer>] [-pushFlag <positive_integer>] [-dropReqWithNoHostHeader (YES | NO)] [-pushEncTriggerTimeout <positive_integer>] [-cryptodevDisableLimit <positive_integer>] [-undefActionControl <string>] [-undefActionData <string>]

Arguments

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

Possible values: 4096, 8192, 16384

Default value: 8192

crlMemorySizeMB

Maximum memory size to use for certificate revocation lists (CRLs). This parameter reserves memory for a CRL but sets a limit to the maximum memory that the CRLs loaded on the appliance can consume.

Default value: 256

Minimum value: 10

Maximum value: 1024

strictCAChecks

Enable strict CA certificate checks on the appliance.

Possible values: YES, NO

Default value: NO

sslTriggerTimeout

Time, in milliseconds, after which encryption is triggered for transactions that are not tracked on the NetScaler appliance because their length is not known. There can be a delay of up to 10ms from the specified timeout value before the packet is pushed into the queue.

Default value: 100

Minimum value: 1

Maximum value: 200

sendCloseNotify

Send an SSL Close-Notify message to the client at the end of a transaction.

Possible values: YES, NO

Default value: YES

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

Default value: 45

Minimum value: 10

Maximum value: 50

denySSLReneg

Deny renegotiation in specified circumstances. Available settings function as follows:

* NO - Allow SSL renegotiation.

* FRONTEND_CLIENT - Deny secure and nonsecure SSL renegotiation initiated by the client.

* FRONTEND_CLIENTSERVER - Deny secure and nonsecure SSL renegotiation initiated by the client or the NetScaler during policy-based client authentication.

* ALL - Deny all secure and nonsecure SSL renegotiation.

* NONSECURE - Deny nonsecure SSL renegotiation. Allows only clients that support RFC 5746.

Possible values: NO, FRONTEND_CLIENT, FRONTEND_CLIENTSERVER, ALL, NONSECURE

Default value: NO

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

Possible values: Unicode, UTF-8

Default value: UNICODE_INSERTION

ocspCacheSize

Size, per packet engine, in megabytes, of the OCSP cache. A maximum of 10% of the packet engine memory can be assigned. Because the maximum allowed packet engine memory is 4GB, the maximum value that can be assigned to

the OCSP cache is approximately 410 MB.

Default value: 10

Maximum value: 512

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 -Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

Maximum value: 3

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

Possible values: YES, NO

Default value: NO

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

Default value: 1

Minimum value: 1

Maximum value: 200

cryptodevDisableLimit

Disabled Crypto Device Limit reboots the system once reached. A value of zero(0) implies no reboot.

undefActionControl

Name of the undefined built-in control action: CLIENTAUTH, NOCLIENTAUTH, NOOP, RESET, or DROP.

Default value: "CLIENTAUTH"

undefActionData

Name of the undefined built-in data action: NOOP, RESET or DROP.

Default value: "NOOP"

unset ssl parameter

Use this command to remove ssl parameter settings.Refer to the set ssl parameter command for meanings of the arguments.

Synopsys

unset ssl parameter [-quantumSize] [-crlMemorySizeMB] [-strictCAChecks] [-sslTriggerTimeout] [-sendCloseNotify] [-encryptTriggerPktCount] [-denySSLReneg] [-insertionEncoding] [-ocspCacheSize] [-pushFlag] [-dropReqWithNoHostHeader] [-pushEncTriggerTimeout] [-cryptodevDisableLimit] [-undefActionControl] [-undefActionData]

show ssl parameter

Displays information about advanced SSL parameters.

Synopsys

show ssl parameter

Arguments

format

level

Outputs

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

crlMemorySizeMB

Maximum memory size to use for certificate revocation lists (CRLs). This parameter reserves memory for a CRL but sets a limit to the maximum memory that the CRLs loaded on the appliance can consume.

strictCAChecks

Memory size to use for CRLs

sslTriggerTimeout

Encryption trigger timer. Set the encryption trigger timeout for transactions, which are not trackable by Netscaler. NetScaler will use this setting to accumulate data received from the server for the configured time period before pushing it to the crypto hardware for encryption.

sendCloseNotify

Send an SSL Close-Notify message to the client at the end of a transaction.

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

denySSLReneg

SSL Renegotiation setting

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

ocspCacheSize

Size, per packet engine, in megabytes of the OCSP cache

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 -Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

cryptodevDisableLimit

Disabled Crypto Device Limit reboots the system once reached. A value of zero(0) implies no reboot

undefActionControl

Global undef action for SSL control policies

undefActionData

Global undef action for SSL data policies

ssl pkcs12

Sep 22, 2015

The following operations can be performed on "ssl pkcs12":

convert ssl pkcs12

Converts the end-user certificate from PEM encoding format to PKCS#12 format. This certificate can then be distributed and installed in browsers as client certificates.

Synopsys

convert ssl pkcs12 <outfile> [-import [-pkcs12File <input_filename>] [-des | -des3]] [-export [-certFile <input_filename>] [-keyFile <input_filename>]] {-password } {-PEMPassPhrase }

Arguments

outfile

Name for and, optionally, path to, the output file that contains the certificate and the private key after converting from PKCS#12 to PEM format. /nsconfig/ssl/ is the default path.

If importing, the certificate-key pair is stored in PEM format. If exporting, the certificate-key pair is stored in PKCS#12 format.

Maximum value: 63

import

Convert the certificate and private-key from PKCS#12 format to PEM format.

export

Convert the certificate and private key from PEM format to PKCS#12 format. On the command line, you are prompted to enter the pass phrase.

password

PEMPassPhrase

Example

1) convert ssl pkcs12 /nsconfig/ssl/client_certkey.p12 -export -cert /nsconfig/ssl/client_certcert.pem -key /nsconfig/ssl/client_key.pem The above example CLI cor

ssl pkcs8

Sep 22, 2015

The following operations can be performed on "ssl pkcs8":

convert ssl pkcs8

Convert a PEM or DER format key file to PKCS#8 format before importing it into the FIPS appliance.

Synopsys

convert ssl pkcs8 <pkcs8File> <keyFile> [-keyform (DER | PEM)] {-password }

Arguments

pkcs8File

Name for and, optionally, path to, the output file where the PKCS#8 format key file is stored. /nsconfig/ssl/ is the default path.

Maximum value: 63

keyFile

Name of and, optionally, path to the input key file to be converted from PEM or DER format to PKCS#8 format. /nsconfig/ssl/ is the default path.

Maximum value: 63

keyform

Format in which the key file is stored on the appliance.

Possible values: DER, PEM

Default value: FORMAT_PEM

password

Password to assign to the file if the key is encrypted. Applies only for PEM format files.

Maximum value: 31

Example

```
convert ssl pkcs8 /nsconfig/ssl/key.pk8 /nsconfig/ssl/key.pem
```

ssl policy

Sep 22, 2015

The following operations can be performed on "ssl policy":

add | rm | set | unset | show

add ssl policy

Adds an SSL policy. An SSL policy evaluates incoming traffic and applies a predefined action to requests that match a rule (expression). You have to configure the actions before creating the policies, so that you can specify an action when you create a policy.

Synopsys

add ssl policy <name> -rule <expression> [-action <string>] [-undefAction <string>] [-comment <string>]

Arguments

name

Name for the new SSL policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

reqAction

The name of the action to be performed on the request. Refer to 'add ssl action' command to add a new action. Builtin actions like NOOP, RESET, DROP, CLIENTAUTH and NOCLIENTAUTH are also allowed.

action

Name of the built-in or user-defined action to perform on the request. Available built-in actions are NOOP, RESET, DROP, CLIENTAUTH, and NOCLIENTAUTH.

undefAction

Name of the action to be performed when the result of rule evaluation is undefined. Possible values for control policies: CLIENTAUTH, NOCLIENTAUTH, NOOP, RESET, DROP. Possible values for data policies: NOOP, RESET or DROP.

comment

Any comments associated with this policy.

Example

```
add ssl action certInsert_act -clientCert ENABLED -certHeader CERT  add ssl policy certInsert_pol -rule 'HTTP.REQ.URL.STARTSWITH("/secure/")' -reqAction certInsert_
```

---

rm ssl policy

Removes an SSL policy.

Synopsys

rm ssl policy <name>

Arguments

name

Name of the SSL policy to be removed.

Example

```
rm ssl policy certInsert_pol
```

set ssl policy

Modifies the parameters of an SSL default syntax policy.

Synopsys

set ssl policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>]

Arguments

name

Name of the SSL policy to modify.

rule

Expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in or user-defined action to perform on the request. Available built-in actions are NOOP, RESET, DROP, CLIENTAUTH, and NOCLIENTAUTH.

undefAction

Name of the action to be performed when the result of rule evaluation is undefined. Possible values for control policies: CLIENTAUTH, NOCLIENTAUTH, NOOP, RESET, DROP. Possible values for data policies: NOOP, RESET or DROP.

comment

Any comments associated with this policy.

Example

```
set ssl policy pol1 -rule "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"
```

unset ssl policy

Removes the attributes of an SSL default syntax policy. Attributes for which a default value is available revert to their default values. Refer to the set ssl policy command for a description of the parameters..Refer to the set ssl policy command for meanings of the arguments.

Synopsys

unset ssl policy <name> [-undefAction] [-comment]

Example

```
unset ssl policy pol1 -undefAction
```

show ssl policy

Displays information about all the SSL policies configured on the appliance, or displays detailed information about the specified SSL policy.

Synopsys

show ssl policy [<name>]

Arguments

name

Name of the SSL policy for which to display detailed information.

summary

fullValues

format

level

Outputs

stateflag

rule

The expression that sets the condition for application of the SSL policy.

action

The name of the action to be performed on the request.

undefAction

Undef Action associated with the policy.

hits

Number of hits for this policy.

piHits

Number of hits.

undefHits

Number of Undef hits.

activePolicy

boundTo

The entity name to which policy is bound

priority

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

comment

Any comments associated with this policy.

bindPolicyType

vserverType

policyType

peFlags

devno

count

Example

show ssl policy 1 SSL policy: 1) Name: certInsert_pol Rule: URL == /* Action: cer

ssl policylabel

Sep 22, 2015

The following operations can be performed on "ssl policylabel":

add | rm | bind | unbind | show

add ssl policylabel

Creates an SSL policy label. An SSL policy label can be a control label or a data label.

Synopsys

add ssl policylabel <labelName> -type (CONTROL | DATA)

Arguments

labelName

Name for the SSL policy label. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my label" or 'my label').

type

Type of policies that the policy label can contain.

Possible values: CONTROL, DATA

Example

```
add ssl policylabel ssl_pol_label -type REQ
```

---

rm ssl policylabel

Removes an SSL policy label.

Synopsys

rm ssl policylabel <labelName>

Arguments

labelName

Name of the SSL policy label to remove.

Example

```
rm ssl policylabel ssl_pol_label
```

---

bind ssl policylabel

Binds an SSL policy to an SSL policy label and specifies the order in which the policies in the label are to be evaluated.

Synopsys

```
bind ssl policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>) ]
```

Arguments

labelName

Name of the SSL policy label to which to bind policies.

policyName

Name of the SSL policy to bind to the policy label.

Example

```
bind ssl policylabel ssl_pol_label -policyName ssl_pol -priority 1
```

---

unbind ssl policylabel

Unbinds an SSL policy from an SSL policy label.

Synopsys

```
unbind ssl policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name of the SSL policy label from which to unbind policies.

policyName

Name of the SSL policy to unbind.

Example

```
unbind ssl policylabel ssl_pol_label ssl_pol
```

---

show ssl policylabel

Displays information about all the SSL policy labels, or displays detailed information about the specified policy label.

Synopsys

show ssl policylabel [<labelName>]

Arguments

labelName

Name of the SSL policy label for which to show detailed information.

summary

fullValues

format

level

Outputs

stateflag

type

Type of policies that the policy label can contain.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the SSL policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke flag.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound SSL policy.

description

Description of the policylabel

flags

devno

count

Example

i) show ssl policylabel ssl_pol_label ii) show ssl policylabel

ssl rsakey

Sep 22, 2015

The following operations can be performed on "ssl rsakey":

create ssl rsakey

Generates an RSA key.

Synopsys

create ssl rsakey <keyFile> <bits> [-exponent (3 | F4)] [-keyform (DER | PEM)] [-des | -des3] {-password }

Arguments

keyFile

Name for and, optionally, path to the RSA key file. /nsconfig/ssl/ is the default path.

Maximum value: 63

bits

Size, in bits, of the RSA key.

Minimum value: 512

Maximum value: 4096

exponent

Public exponent for the RSA key. The exponent is part of the cipher algorithm and is required for creating the RSA key.

Possible values: 3, F4

Default value: FIPSEXP_F4

keyform

Format in which the RSA key file is stored on the appliance.

Possible values: DER, PEM

Default value: FORMAT_PEM

des

Encrypt the generated RSA key by using the DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that is used to encrypt the key.

des3

Encrypt the generated RSA key by using the Triple-DES algorithm. On the command line, you are prompted to enter

the pass phrase (password) that is used to encrypt the key.

password

Pass phrase to use for encryption if DES or DES3 option is selected.

Maximum value: 31

Example

```
create ssl rsakey /nsconfig/ssl/rsa1024.pem 1024 -exp F4
```

ssl service

Sep 22, 2015

The following operations can be performed on "ssl service":

set | unset | bind | unbind | show

set ssl service

Sets the advanced SSL configuration for an SSL service.

Synopsys

set ssl service <serviceName>@ [-dh (ENABLED | DISABLED) -dhFile <string>] [-dhCount <positive_integer>] [-eRSA (ENABLED | DISABLED) [-eRSACount <positive_integer>]] [-sessReuse (ENABLED | DISABLED) [-sessTimeout <positive_integer>]] [-cipherRedirect (ENABLED | DISABLED) [-cipherURL <URL>]] [-sslv2Redirect (ENABLED | DISABLED) [-sslv2URL <URL>]] [-clientAuth (ENABLED | DISABLED) [-clientCert (Mandatory | Optional)]] [-sslRedirect (ENABLED | DISABLED)] [-redirectPortRewrite (ENABLED | DISABLED)] [-nonFipsCiphers (ENABLED | DISABLED)] [-ssl2 (ENABLED | DISABLED)] [-ssl3 (ENABLED | DISABLED)] [-tls1 (ENABLED | DISABLED)] [-tls11 (ENABLED | DISABLED)] [-tls12 (ENABLED | DISABLED)] [-SNIEnable (ENABLED | DISABLED)] [-serverAuth (ENABLED | DISABLED) [-commonName <string>]] [-pushEncTrigger <pushEncTrigger>] [-sendCloseNotify (YES | NO)] [-dtlsProfileName <string>]

Arguments

serviceName

Name of the SSL service.

dh

State of Diffie-Hellman (DH) key exchange. This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dhCount

Number of interactions, between the client and the NetScaler appliance, after which the DH private-public pair is regenerated. A value of zero (0) specifies infinite use (no refresh). This parameter is not applicable when configuring a backend service.

Maximum value: 65534

eRSA

State of Ephemeral RSA (eRSA) key exchange. Ephemeral RSA allows clients that support only export ciphers to communicate with the secure server even if the server certificate does not support export clients. The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted. It is reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the appliance restarts.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

cipherRedirect

State of Cipher Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a cipher mismatch between the virtual server or service and the client.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

sslv2Redirect

State of SSLv2 Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a protocol version mismatch between the virtual server or service and the client.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientAuth

State of client authentication. In service-based SSL offload, the service terminates the SSL handshake if the SSL client does not provide a valid certificate.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

sslRedirect

State of HTTPS redirects for the SSL service.

For an SSL session, if the client browser receives a redirect message, the browser tries to connect to the new location. However, the secure SSL session breaks if the object has moved from a secure site (https://) to an unsecure site (http://). Typically, a warning message appears on the screen, prompting the user to continue or disconnect.

If SSL Redirect is ENABLED, the redirect message is automatically converted from http:// to https:// and the SSL session does not break.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectPortRewrite

State of the port rewrite while performing HTTPS redirect. If this parameter is set to ENABLED, and the URL from the server does not contain the standard port, the port is rewritten to the standard.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl2

State of SSLv2 protocol support for the SSL service.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls11

State of TLSv1.1 protocol support for the SSL service.Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls12

State of TLSv1.2 protocol support for the SSL service.Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

SNIEnable

State of the Server Name Indication (SNI) feature on the virtual server and service-based offload. SNI helps to enable SSL encryption on multiple domains on a single virtual server or service if the domains are controlled by the same organization and share the same second-level domain name. For example, *.sports.net can be used to secure domains such as login.sports.net and help.sports.net.

Possible values: ENABLED, DISABLED

Default value: DISABLED

serverAuth

State of server authentication support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushEncTrigger

Trigger encryption on the basis of the PUSH flag value. Available settings function as follows:

* ALWAYS - Any PUSH packet triggers encryption.

* IGNORE - Ignore PUSH packet for triggering encryption.

* MERGE - For a consecutive sequence of PUSH packets, the last PUSH packet triggers encryption.

* TIMER - PUSH packet triggering encryption is delayed by the time defined in the set ssl parameter command or in the Change Advanced SSL Settings dialog box.

Possible values: Always, Merge, Ignore, Timer

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

dtlsProfileName

Name of the DTLS profile whose settings are to be applied to the virtual server.

Example

1) set ssl service sslsvc -dh ENABLED -dhFile /nsconfig/ssl/dh1024.pem -dhCount 500 The above example sets the DH parameters for the SSL service 'sslsvc'. 2. set ssl se

---

unset ssl service

Use this command to remove ssl service settings.Refer to the set ssl service command for meanings of the arguments.

Synopsys

unset ssl service <serviceName>@ [-dh] [-dhFile] [-dhCount] [-eRSA] [-eRSACount] [-sessReuse] [-sessTimeout] [-cipherRedirect] [-cipherURL] [-sslv2Redirect] [-sslv2URL] [-clientAuth] [-clientCert] [-sslRedirect] [-redirectPortRewrite] [-nonFipsCiphers] [-ssl2] [-ssl3] [-tls1] [-tls11] [-tls12] [-SNIEnable] [-serverAuth] [-commonName] [-sendCloseNotify] [-dtlsProfileName]

---

bind ssl service

Binds an SSL certificate-key pair or an SSL policy to a transparent SSL service.

Synopsys

bind ssl service <serviceName>@ ((-policyName <string> [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)]) | ((-certkeyName <string> [(-CA [-crlCheck (Mandatory | Optional) | -ocspCheck (Mandatory | Optional)] [-skipCAName]) | -SNICert]) | -cipherName <string>))

Arguments

serviceName

Name of the SSL service for which to set advanced configuration.

policyName

Name of the SSL policy to bind to the service.

certkeyName

Name of the certificate-key pair.

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias.

Example

bind ssl service ssl_svc -policyName certInsert_pol -priority 10

---

unbind ssl service

Unbinds an SSL policy, cipher, and certificate-key pair from an SSL service.

Synopsys

unbind ssl service <serviceName>@ ((-policyName <string> [-priority <positive_integer>]) | ((-certkeyName <string> [(-CA [-crlCheck (Mandatory | Optional)]) | -SNICert]) | -cipherName <string>))

Arguments

serviceName

Name of the SSL service.

policyName

Name of the SSL policy to unbind from the SSL service.

certkeyName

The certificate key pair binding.

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias.

Example

```
unbind ssl service ssl_svc -policyName certInsert_pol
```

---

show ssl service

Displays information about SSL-specific configuration information for all SSL services, or displays detailed information about the specified SSL service.

Synopsys

show ssl service [<serviceName>] [-cipherDetails]

Arguments

serviceName

Name of the SSL service for which to show detailed information.

cipherDetails

Display details of the individual ciphers bound to the SSL service.

summary

fullValues

format

level

Outputs

crlCheck

The state of the CRL check parameter. (Mandatory/Optional)

dh

The state of Diffie-Hellman (DH) key exchange support.

dhFile

The file name and path for the DH parameter.

dhCount

The refresh count for regeneration of DH public-key and private-key from the DH parameter.

eRSA

The state of Ephemeral RSA key exchange support. Ephemeral RSA is used for export ciphers

eRSACount

The refresh count for re-generation of RSA public-key and pri-vate-key pair.

sessReuse

The state of session reuse support.

sessTimeout

The session timeout value in seconds.

cipherRedirect

The state of Cipher Redirect feature.Cipher Redirect feature can be used to provide more readable information to SSL clients about mismatch in ciphers between

the client and the SSL vserver.

cipherURL

The redirect URL to be used with the Cipher Redirect feature.

sslv2Redirect

The state of SSLv2 Redirect feature.SSLv2 Redirect feature can be used to provide more readable information to SSL client about non-support of SSLv2 protocol on the SSL vserver.

sslv2URL

The redirect URL to be used with the SSLv2 Redirect feature.

clientAuth

The state of Client-Authentication support.

clientCert

The rule for client certificate requirement in client authentication.

sslRedirect

The state of HTTPS redirect feature.

redirectPortRewrite

The state of port rewrite feature.

nonFipsCiphers

The state of usage of non FIPS approved ciphers.

ssl2

The state of SSLv2 protocol support.

ssl3

The state of SSLv3 protocol support.

tls1

The state of TLSv1.0 protocol support.

tls11

The state of TLSv1.1 protocol support.

tls12

The state of TLSv1.2 protocol support.

SNIEnable

The state of SNI extension. Server Name Indication (SNI) helps to enable SSL encryption on multiple subdomains if the domains are controlled by the same organization and share the same second-level domain name.

serverAuth

The state of Server-Authentication support.

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

cipherAliasName/cipherName/cipherGroupName

The cipher group/alias/individual cipher configuration.

cipherName

The cipher group/alias/individual cipher configuration

description

The cipher suite description.

certkeyName

The certificate key pair binding.

policyName

The SSL policy binding.

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

clearTextPort

The clearTextPort settings.

service

priority

The priority of the policies bound to this SSL service

polinherit

Whether the bound policy is a inherited policy or not

ocspCheck

Rule to use for the OCSP responder associated with the CA certificate during client authentication. If MANDATORY is specified, deny all SSL clients if the OCSP check fails because of connectivity issues with the remote OCSP server, or any other reason that prevents the OCSP check. With the OPTIONAL setting, allow SSL clients even if the OCSP check fails except when the client certificate is revoked.

pushEncTrigger

PUSH packet triggering encryption: Always, Ignore, Merge

CA

CA certificate.

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

stateflag

skipCAName

The flag is used to indicate whether this particular CA certificate's CA_Name needs to be sent to the SSL client while requesting for client certificate in a SSL handshake

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

dtlsProfileName

Name of the DTLS profile whose settings are to be applied to the virtual server.

dtlsFlag

The flag is used to indicate whether DTLS is set or not

devno

count

Example

An example of output of show ssl service command is as shown below show ssl service svc1 Advanced SSL configuration f

ssl serviceGroup

Sep 22, 2015

The following operations can be performed on "ssl serviceGroup":

set | unset | bind | unbind | show

set ssl serviceGroup

Sets the advanced SSL configuration for an SSL service group.

Synopsys

set ssl serviceGroup <serviceGroupName>@ [-sessReuse (ENABLED | DISABLED) [-sessTimeout <positive_integer>]] [-nonFipsCiphers (ENABLED | DISABLED)] [-ssl3 (ENABLED | DISABLED)] [-tls1 (ENABLED | DISABLED)] [-serverAuth (ENABLED | DISABLED) [-commonName <string>]] [-sendCloseNotify (YES | NO)]

Arguments

serviceGroupName

Name of the SSL service group for which to set advanced configuration.

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

serverAuth

State of server authentication support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: DISABLED

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

Example

1) set ssl servicegroup svcg1 -sessReuse DISABLED The above example disables session reuse for the service group 'svcg1'.

---

unset ssl serviceGroup

Use this command to remove ssl serviceGroup settings.Refer to the set ssl serviceGroup command for meanings of the arguments.

Synopsys

unset ssl serviceGroup <serviceGroupName>@ [-sessReuse] [-sessTimeout] [-nonFipsCiphers] [-ssl3] [-tls1] [-serverAuth] [-commonName] [-sendCloseNotify]

---

bind ssl serviceGroup

Bind a SSL certkey or a SSL policy to a SSL service.

Synopsys

bind ssl serviceGroup <serviceGroupName>@ ((-certkeyName <string> [(-CA [-crlCheck (Mandatory | Optional) | -ocspCheck (Mandatory | Optional)]) | -SNICert]) | -cipherName <string>)

Arguments

serviceGroupName

The name of the SSL service to which the SSL policy needs to be bound.

certkeyName

The name of the CertKey

cipherName

A cipher-suite can consist of an individual cipher name, the system predefined cipher-alias name, or user defined cipher-group name.

Example

```
bind ssl service ssl_svc -policyName certInsert_pol -priority 10
```

---

unbind ssl serviceGroup

Unbind a SSL policy from a SSL service.

Synopsys

unbind ssl serviceGroup <serviceGroupName>@ ((-certkeyName <string> [(-CA [-crlCheck (Mandatory | Optional)]) | -SNICert]) | -cipherName <string>)

Arguments

serviceGroupName

The name of the SSL service from which the SSL policy needs to be unbound.

certkeyName

The name of the certificate bound to the SSL service group.

cipherName

A cipher-suite can consist of an individual cipher name, the system predefined cipher-alias name, or user defined cipher-group name.

Example

```
unbind ssl service ssl_svc -policyName certInsert_pol
```

---

show ssl serviceGroup

Displays information about SSL-specific configuration for all SSL service groups, or displays detailed information about the specified SSL service group.

Synopsys

show ssl serviceGroup [<serviceGroupName>] [-cipherDetails]

Arguments

serviceGroupName

Name of the SSL service group for which to show detailed information.

cipherDetails

Display details of the individual ciphers bound to the SSL service group.

summary

fullValues

format

level

Outputs

dh

The state of DH key exchange support for the SSL service group.

dhFile

The file name and path for the DH parameter.

dhCount

The refresh count for the re-generation of DH public-key and private-key from the DH parameter.

eRSA

The state of Ephemeral RSA key exchange support for the SSL service group.Ephemeral RSA is used for export ciphers.

eRSACount

The refresh count for the re-generation of RSA public-key and private-key pair.

sessReuse

The state of session re-use support for the SSL service group.

sessTimeout

The Session timeout value in seconds.

cipherRedirect

The state of Cipher Redirect feature. Cipher Redirect feature can be used to provide more readable information to SSL clients about mismatch in ciphers between the client and the SSL vserver.

cipherURL

The redirect URL to be used with the Cipher Redirect feature.

sslv2Redirect

The state of SSLv2 Redirect feature.SSLv2 Redirect feature can be used to provide more readable information to SSL client about non-support of SSLv2 protocol on the SSL vserver.

sslv2URL

The redirect URL to be used with SSLv2 Redirect feature.

clientAuth

The state of Client-Authentication support for the SSL service group.

clientCert

The rule for client certificate requirement in client authentication.

sslRedirect

The state of HTTPS redirects for the SSL service group.

This is required for the proper functioning of the redirect messages from the server. The redirect message from the server provides the new location for the moved object. This is contained in the HTTP header field: Location, e.g. Location: http://www.moved.org/here.html

For the SSL session, if the client browser receives this message, the browser will try to connect to the new location. This will break the secure SSL session, as the object has moved from a secure site (https://) to an un-secure one (http://). Generally browsers flash a warning message on the screen and prompt the user, either to continue or disconnect.

The above feature, when enabled will automatically convert all such http:// redirect message to https://. This will not break the client SSL session.

Note: The set ssl service command can be used for configuring a front-end SSL service for service based SSL Off-Loading, or a backend SSL service for backend-encryption setup.

redirectPortRewrite

The state of port-rewrite feature.

nonFipsCiphers

The state of usage of non FIPS approved ciphers.

ssl2

The state of SSLv2 protocol support for the SSL service group.

ssl3

State of SSLv3 protocol support for the SSL service group.

tls1

State of TLSv1.0 protocol support for the SSL service group.

serverAuth

The state of the server authentication configuration for the SSL service group. For SSL deployments where data is encrypted end-to-end using SSL, you can authenticate the server.

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

cipherAliasName/cipherName/cipherGroupName

The name of the cipher group/alias/name configured for the SSL service group.

cipherName

The name of the cipher group/alias/name configured for the SSL service group.

ocspCheck

The state of the OCSP check parameter. (Mandatory/Optional)

crlCheck

The state of the CRL check parameter. (Mandatory/Optional)

description

The description of the cipher.

certkeyName

The name of the certificate bound to the SSL service group.

clearTextPort

The port on the back-end web-servers where the clear-text data is sent by system. Use this setting for the wildcard IP based SSL Acceleration configuration (*:443).

serviceName

The service name.

CA

CA certificate.

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

stateflag

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

devno

count

Example

An example of output of show ssl servicegroup command is as shown below show ssl servicegroup ssl_svcg Advanced SSL configu

1) set ssl vserver sslvip -dh ENABLED -dhFile /siteA/dh1024.pem -dhCount 500 The above example set the DH parameters for the SSL virtual server 'sslvip'. 3) set ssl vser

1. bind ssl vserver ssl_vip -certkeyName cert1 In the above example the certificate cert1 is bound to the SSL vserver ssl_vip as server certificate. 2. bind ssl vserver ssl_v

```
unbind ssl vserver ssl_vip -policyName certInsert_pol
```

An example of the output of the show vserver sslvip command is as follows: sh ssl vserver va1 Advanced SSL configuration for VS

```
create wrapkey wrap1 -password wrapkey123 -salt wrapsalt123
```

```
rm wrapkey wrap1
```

An example of output of 'show wrapkey' command is as shown below: sh wrapkey 1 WRAP key: 1) WRAP Key Name: wrap1

-
-
-

```
add stream identifier stream_id top_url -interval 10 -sampleCount 1 -sort REQUESTS
```

```
set stream identifier stream_id -selectorName top_clients -interval 1 -sampleCount 1 -sort NONE
```

```
rm stream identifier stream_id
```

```
show stream identifier stream_id
```

```
add stream selector sel_subnet HTTP.REQ.URL CLIENT.IP.SRC.SUBNET(24)
```

```
set stream sel_subnet HTTP.REQ.URL CLIENT.IP.SRC
```

```
rm stream selector sel_subnet
```

```
show ns limitSelector sel_subnet
```

```
clear stream session stream_id
```

-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-

```
show system entity lbvserver
```

```
show system entitydata lbvserver v1 totalrequests -last 1 days
```

system eventhistory

Sep 22, 2015

The following operations can be performed on "system eventhistory":

Display events in historical data.

show system eventhistory [-startTime <string> | (-last <integer> [<unit>])] [-endTime <string>] -dataSource <string>

startTime

Specify start time in mmddyyyyhhmm to start collecting values from that timestamp.

endTime

Specify end time in mmddyyyyhhmm upto which values have to be collected.

last

Last is literal way of saying a certain time period from the current moment. Example: -last 1 hour, -last 1 day, et cetera.

Default value: 1

dataSource

Specifies the source which contains all the stored counter values.

response

system global

Sep 22, 2015

The following operations can be performed on "system global":

bind | unbind | show

Binds policies globally.

bind system global [<policyName> [-priority <positive_integer>]]

policyName

Name of the policy to bind globally.

Unbinds a globally bound policy.

unbind system global <policyName>

policyName

Name of the globally bound policy to unbind.

Displays information about all global policy bindings.

show system global

summary

fullValues

format

level

policyName

The name of the command policy.

priority

The priority of the command policy.

bindPolicyType

Bound policy type

policySubType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

stateflag

devno

count

system globaldata

Sep 22, 2015

The following operations can be performed on "system globaldata":

Display historical data for global counters.

show system globaldata <counters> [<countergroup>] [-startTime <string> | (-last <integer> [<unit>])] [-endTime <string>] [-dataSource <string>] [-core <integer>]

counters

Specify the counters to be collected.

countergroup

Specify the (counter) group name which contains all the counters specific to this particular group.

startTime

Specify start time in mmddyyyyhhmm to satrt collecting values from that timestamp.

endTime

Specify end time in mmddyyyyhhmm upto which values have to be collected.

last

Last is literal way of saying a certain time period from the current moment. Example: -last 1 hour, -last 1 day, et cetera.

Default value: 1

dataSource

Specifies the source which contains all the stored counter values.

core

Specify core ID of the PE in nCore.

response

startUpdate

lastupdate

```
show system globaldata cpu_usage -last 1 hours
```

system group

Sep 22, 2015

The following operations can be performed on "system group":

add | rm | bind | unbind | show | set | unset

Creates a system-user group, to which you can bind individual users by using the bind system group command.

add system group <groupName> [-promptString <string>] [-timeout <secs>]

groupName

Name for the group. Must begin with a letter, number, or the underscore (_) character, and must contain only alphanumeric, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), and underscore characters. Cannot be changed after the group is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my group" or 'my group').

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

Maximum value: 100000000

Removes a system group from the appliance.

rm system group <groupName>

groupName

Name of the system group to remove.

Binds a system user to a system group.

bind system group <groupName> [-userName <string>] [-policyName <string> <priority>]

groupName

Name of the system group.

userName

Name of a system user to bind to the group.

policyName

Name of the command policy to be bind to the group.

Unbinds a system user from a group.

unbind system group <groupName> [-userName <string>] [-policyName <string>]

groupName

Name of the system group from which to unbind the user.

userName

Name of the system user to unbind from the group.

policyName

Command policy to unbind from the group.

Displays information about all system groups configured on the appliance, or about the specified group.

show system group [<groupName>]

groupName

Name of the system group about which to display information.

summary

fullValues

format

level

userName

The system user.

policyName

The name of command policy.

priority

The priority of the command policy.

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

devno

count

stateflag

Modifies the specified parameters of a system group.

set system group <groupName> [-promptString <string>] [-timeout <secs>]

groupName

Name of system group to be modified.

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

Maximum value: 100000000

Use this command to remove system group settings.Refer to the set system group command for meanings of the arguments.

unset system group <groupName> [-promptString] [-timeout]

---

system memory

Sep 22, 2015

The following operations can be performed on "system memory":

Displays system-memory statistics.

stat system memory [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

clearstats

Clear the statsistics / counters

Possible values: basic, full

Shared Memory InUse (%) (shMemAllocpcnt)

Shared memory insue percent.

Shared Memory InUse (MB) (shMemAllocMB)

Shared memory insue, in megabytes.

Total Shared Memory (MB) (shMemtotMB)

Total shared memory allowed to allocate, in megabytes.

Free Memory (MB) (MemTotFree)

Total Free PE Memory in the System.

InUse Memory (%) (MemUsage)

Percentage of memory utilization on NetScaler.

InUse Memory (MB) (MemTotUseMB)

Total NetScaler Memory in use, in megabytes.

Memory Allocated (%) (MemTotAlloc(%))

Currently allocated memory in percent.

Memory Allocated (MB) (MemTotAlloc)

Currently allocated memory, in megabytes.

Memory Currently Available (MB) (MemTotMB)

Total memory available (grabbed) for use by packet engine (PE), in megabytes.

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

stat system memory

system parameter

Sep 22, 2015

The following operations can be performed on "system parameter":

set | unset | show

Modifies the specified system parameters.

set system parameter [-rbaOnResponse (ENABLED | DISABLED)] [-promptString <string>] [-natPcbForceFlushLimit <positive_integer>] [-natPcbRstOnTimeout (ENABLED | DISABLED)] [-timeout <secs>]

rbaOnResponse

Enable or disable Role-Based Authentication (RBA) on responses.

Possible values: ENABLED, DISABLED

Default value: ENABLED

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

natPcbForceFlushLimit

Flush the system if the number of Network Address Translation Protocol Control Blocks (NATPCBs) exceeds this value.

Default value: 2147483647

Minimum value: 1000

natPcbRstOnTimeout

Send a reset signal to client and server connections when their NATPCBs time out. Avoids the buildup of idle TCP connections on both the sides.

Possible values: ENABLED, DISABLED

Default value: DISABLED

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

Maximum value: 100000000

Use this command to remove system parameter settings.Refer to the set system parameter command for meanings of the arguments.

unset system parameter [-rbaOnResponse] [-promptString] [-natPcbForceFlushLimit] [-natPcbRstOnTimeout] [-timeout]

Displays information about the system parameters.

show system parameter

format

level

rbaOnResponse

Enable or disable Role-Based Authentication (RBA) on responses.

promptString

The global system prompt.

natPcbForceFlushLimit

Flush the system if the number of Network Address Translation Protocol Control Blocks (NATPCBs)

exceeds this value.

natPcbRstOnTimeout

Send RST to client and server connections when the natpcbs timeout.This avoids the buildup of idle TCP connections on the both the sides.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

maxClient

Maximum number of client connection allowed by the system

system session

Sep 22, 2015

The following operations can be performed on "system session":

show | kill

Displays information about all current system sessions, or about the specified session. The system might reclaim sessions with no active connections before expiry time.

show system session [<sid>]

sid

ID of the system session about which to display information.

Minimum value: 1

summary

fullValues

userName

user name of the session

logintime

logged-in time of this session

lastactivitytime

last activity time of on this session

expirytime

Time left in expire the session in seconds

numOfconnections

number of connection using this token

currentconn

True if the token is used for current session

devno

count

stateflag

Kills one system session, or all system sessions except the current session.

kill system session (<sid> | -all)

sid

ID of the system session to terminate.

CLI users: You can get the session ID by using the show system session command.

Minimum value: 1

all

Terminate all the system sessions except the current session.

---

system user

Sep 22, 2015

The following operations can be performed on "system user":

add | rm | set | unset | bind | unbind | show

Adds a new user to the system. Note: You must provide the password after the user name.

add system user <userName> [-externalAuth (ENABLED | DISABLED)] [-promptString <string>] [-timeout <secs>]

userName

Name for a user. Must begin with a letter, number, or the underscore (_) character, and must contain only alphanumeric, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), and underscore characters. Cannot be changed after the user is added.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my user" or 'my user').

password

Password for the system user. Can include any ASCII character.

externalAuth

Whether to use external authentication servers for the system user authentication or not

Possible values: ENABLED, DISABLED

Default value: ENABLED

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

Maximum value: 100000000

Removes a system user from the appliance.

```
rm system user <userName>
```

userName

Name of the system user to remove.

Modifies the specified parameters of a system-user entry.

```
set system user <userName> {-password } [-externalAuth ( ENABLED | DISABLED )] [-promptString <string>] [-timeout <secs>]
```

userName

Name of the system-user entry to modify.

password

Password for the system user. Can include any ASCII character.

externalAuth

Whether to use external authentication servers for the system user authentication or not

Possible values: ENABLED, DISABLED

Default value: ENABLED

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

Maximum value: 100000000

Use this command to remove system user settings.Refer to the set system user command for meanings of the arguments.

```
unset system user <userName> [-externalAuth] [-promptString] [-timeout]
```

Binds a command policy to a system user.

```
bind system user <userName> <policyName> <priority>
```

userName

Name of the system-user entry to which to bind the command policy.

policyName

Name of the command policy to bind to the system user.

Unbinds a command policy from the system user.

unbind system user <userName> <policyName>

userName

Name of the user entry from which to unbind the command policy.

policyName

Name of the command policy to unbind.

Displays information about all system users configured on the appliance, or about the specified user.

show system user [<userName>]

userName

Name of a system user about whom to display information.

summary

fullValues

format

level

groupName

The system group.

policyName

The name of command policy.

priority

The priority of the policy.

password

Password for the system user. Can include any ASCII character.

encrypted

externalAuth

Whether to use external authentication servers for the system user authentication or not

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space ( ), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

promptInheritedFrom

From where the prompt has been inherited.

timeout

CLI session inactivity timeout, in seconds. Timeout cannot have values in between 1 and 9.

timeoutKind

From where the timeout has been inherited.

devno

count

stateflag

Traffic Management Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- tm formSSOAction
- tm global
- tm samlSSOProfile
- tm sessionAction
- tm sessionParameter
- tm sessionPolicy
- tm trafficAction
- tm trafficPolicy

tm formSSOAction

Sep 22, 2015

The following operations can be performed on "tm formSSOAction":

add | rm | set | unset | show

Creates a form-based single sign-on traffic profile (action.) Form-based single sign-on allows users to access web applications that require an HTML form-based logon without having to type their password again for each new application.

add tm formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule <expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)]

name

Name for the new form-based single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

actionURL

URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Expression, that checks to see if single sign-on is successful.

nameValuePair

Name-value pair attributes to send to the server in addition to sending the username and password. Value names are separated by an ampersand (&) (for example, name1=value1&name2=value2).

responsesize

Number of bytes, in the response, to parse for extracting the forms.

Default value: 8096

nvtype

Type of processing of the name-value pair. If you specify STATIC, the values configured by the administrator are used. For DYNAMIC, the response is parsed, and the form is extracted and then submitted.

Possible values: STATIC, DYNAMIC

Default value: NS_ACT_FSSO_NV_DYNAMIC

submitMethod

HTTP method used by the single sign-on form to send the logon credentials to the logon server. Applies only to STATIC name-value type.

Possible values: GET, POST

Default value: NS_ACT_FSSO_SUBMIT_GET

Deletes an existing form-based single sign-on traffic profile (action.)

rm tm formSSOAction <name>

name

Name of the form-based single sign-on profile to delete.

Modifies the specified attributes of a form-based single sign-on traffic profile (action.)

set tm formSSOAction <name> [-actionURL <URL>] [-userField <string>] [-passwdField <string>] [-ssoSuccessRule <expression>] [-responsesize <positive_integer>] [-nameValuePair <string>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)]

name

Name of the form-based single sign-on profile (action) to modify.

actionURL

URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Expression, that checks to see if single sign-on is successful.

responsesize

Number of bytes, in the response, to parse for extracting the forms.

Default value: 8096

nameValuePair

Name-value pair attributes to send to the server in addition to sending the username and password. Value names are separated by an ampersand (&) (for example, name1=value1&name2=value2).

nvtype

Type of processing of the name-value pair. If you specify STATIC, the values configured by the administrator are used. For DYNAMIC, the response is parsed, and the form is extracted and then submitted.

Possible values: STATIC, DYNAMIC

Default value: NS_ACT_FSSO_NV_DYNAMIC

submitMethod

HTTP method used by the single sign-on form to send the logon credentials to the logon server. Applies only to STATIC name-value type.

Possible values: GET, POST

Default value: NS_ACT_FSSO_SUBMIT_GET

Use this command to remove tm formSSOAction settings.Refer to the set tm formSSOAction command for meanings of the arguments.

unset tm formSSOAction <name> [-responsesize] [-nameValuePair] [-nvtype] [-submitMethod]

Displays information about all configured form-based single sign-on actions, or displays detailed information about the specified action.

show tm formSSOAction [<name>]

name

Name of the SSO action for which to display detailed information.

summary

fullValues

format

level

actionURL

URL to which the completed form is submitted.

userField

Username field.

passwdField

Password field.

responsesize

Number of bytes, in the response, to parse for extracting the forms.

nameValuePair

Form attributes and their values to be submitted.

nvtype

Bypass Form extraction

ssoSuccessRule

Rule to be evaluated to check whether sso succeeded or not.

submitMethod

Form Submit method.

devno

count

stateflag

tm global

Sep 22, 2015

The following operations can be performed on "tm global":

bind | unbind | show

Binds traffic, sessions, nslog, and syslog policies to traffic management (TM) Global.

bind tm global [-policyName <string> [-priority <positive_integer>]]

policyName

Name of the policy that you are binding.

Unbinds a globally bound traffic session policy.

unbind tm global -policyName <string>

policyName

Name of the policy to unbind.

Displays information about TM global bindings.

show tm global

summary

fullValues

format

level

policyName

The name of the policy.

priority

The priority of the policy.

type

Bindpoint to which the policy is bound

policySubType

stateflag

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

bindPolicyType

Bound policy type

devno

count

tm samlSSOProfile

Sep 22, 2015

The following operations can be performed on "tm samlSSOProfile":

add | rm | set | unset | show

Creates a SAML single sign-on profile. This profile is employed in triggering saml assertion to a target service based on traffic profile.

add tm samlSSOProfile <name> -samlSigningCertName <string> -assertionConsumerServiceURL <URL> -relaystateRule <expression> [-sendPassword (ON | OFF)] [-samlIssuerName <string>]

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

relaystateRule

Expression to extract relaystate to be sent along with assertion.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

Deletes an existing saml single sign-on traffic profile.

rm tm samlSSOProfile <name>

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

Modifies the specified attributes of a saml single sign-on traffic profile.

set tm samlSSOProfile <name> [-samlSigningCertName <string>] [-assertionConsumerServiceURL <URL>] [-sendPassword (ON | OFF)] [-samlIssuerName <string>] [-relaystateRule <expression>]

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion.

Use this command to remove tm samlSSOProfile settings.Refer to the set tm samlSSOProfile command for meanings of the arguments.

unset tm samlSSOProfile <name> [-samlSigningCertName] [-sendPassword] [-samlIssuerName]

Displays information about all configured saml single sign-on profiles, or displays detailed information about the specified action.

show tm samlSSOProfile [<name>]

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

summary

fullValues

format

level

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion.

devno

count

stateflag

tm sessionAction

Sep 22, 2015

The following operations can be performed on "tm sessionAction":

add | rm | set | unset | show

Creates a session action (profile) that allows you to override global settings for any of the session parameters.

add tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-httpOnlyCookie (YES | NO)] [-kcdAccount <string>] [-persistentCookie (ON | OFF)] [-persistentCookieValidity <mins>] [-homePage <URL>]

name

Name for the session action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after a session action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

sessTimeout

Session timeout, in minutes. If there is no traffic during the timeout period, the user is disconnected and must reauthenticate to access intranet resources.

Minimum value: 1

defaultAuthorizationAction

Allow or deny access to content for which there is no specific authorization policy.

Possible values: ALLOW, DENY

SSO

Use single sign-on (SSO) to log users on to all web applications automatically after they authenticate, or pass users to the web application logon page to authenticate to each application individually.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Use the primary or secondary authentication credentials for single sign-on (SSO).

Possible values: PRIMARY, SECONDARY

ssoDomain

Domain to use for single sign-on (SSO).

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

Possible values: YES, NO

kcdAccount

Kerberos constrained delegation account name

persistentCookie

Enable or disable persistent SSO cookies for the traffic management (TM) session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends. This setting is overwritten if a traffic action sets persistent cookie to OFF.

Note: If persistent cookie is enabled, make sure you set the persistent cookie validity.

Possible values: ON, OFF

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistent cookie setting is enabled.

Minimum value: 1

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

Deletes an existing session action.

rm tm sessionAction <name>

name

Name of the session action to delete.

Modifies the specified parameters of an existing session action.

set tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-kcdAccount <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ON | OFF)] [-persistentCookieValidity <positive_integer>] [-homePage <URL>]

name

Name of the session action to modify.

sessTimeout

Session timeout, in minutes. If there is no traffic during the timeout period, the user is disconnected and must reauthenticate to access intranet resources.

Minimum value: 1

defaultAuthorizationAction

Allow or deny access to content for which there is no specific authorization policy.

Possible values: ALLOW, DENY

SSO

Use single sign-on (SSO) to log users on to all web applications automatically after they authenticate, or pass users to the web application logon page to authenticate to each application individually.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Use the primary or secondary authentication credentials for single sign-on (SSO).

Possible values: PRIMARY, SECONDARY

ssoDomain

Domain to use for single sign-on (SSO).

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

Possible values: YES, NO

persistentCookie

Enable or disable persistent SSO cookies for the traffic management (TM) session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends. This setting is overwritten if a traffic action sets persistent cookie to OFF.

Note: If persistent cookie is enabled, make sure you set the persistent cookie validity.

Possible values: ON, OFF

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistent cookie setting is enabled.

Minimum value: 1

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

Use this command to remove tm sessionAction settings.Refer to the set tm sessionAction command for meanings of the arguments.

unset tm sessionAction <name> [-sessTimeout] [-defaultAuthorizationAction] [-SSO] [-ssoCredential] [-ssoDomain] [-kcdAccount] [-httpOnlyCookie] [-persistentCookie] [-persistentCookieValidity] [-homePage]

Displays information about all configured traffic management (TM) session actions, or detailed information about the specified TM session action.

show tm sessionAction [<name>]

name

Name of the existing traffic management (TM) session action for which to display detailed information.

summary

fullValues

format

level

sessTimeout

The session timeout, in minutes, set by the action.

defaultAuthorizationAction

The Authorization Action, e.g. allow or deny

stateflag

SSO

Whether or not Single Sign-On is used for this session.

ssoCredential

Use the primary or secondary authentication credentials for single sign-on (SSO).

ssoDomain

Domain to use for single sign-on (SSO).

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

persistentCookie

Enable or disable persistent SSO cookies for the traffic management (TM) session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends. This setting is overwritten if a traffic action sets persistent cookie to OFF.

Note: If persistent cookie is enabled, make sure you set the persistent cookie validity.

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistent cookie setting is enabled.

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

tm sessionParameter

Sep 22, 2015

The following operations can be performed on "tm sessionParameter":

set | unset | show

Sets global parameters for the traffic management (TM) session. Parameters defined when adding a traffic session action override these parameters.

set tm sessionParameter [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-kcdAccount <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ON | OFF)] [-persistentCookieValidity <positive_integer>] [-homePage <URL>]

sessTimeout

Session timeout, in minutes. If there is no traffic during the timeout period, the user is disconnected and must reauthenticate to access the intranet resources.

Default value: 30

Minimum value: 1

defaultAuthorizationAction

Allow or deny access to content for which there is no specific authorization policy.

Possible values: ALLOW, DENY

Default value: NS_ALLOW

SSO

Log users on to all web applications automatically after they authenticate, or pass users to the web application logon page to authenticate for each application.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Use primary or secondary authentication credentials for single sign-on.

Possible values: PRIMARY, SECONDARY

Default value: VPN_SESS_ACT_USE_PRIMARY_CREDENTIALS

ssoDomain

Domain to use for single sign-on.

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

Possible values: YES, NO

Default value: VPN_SESS_ACT_HTTPONLYCOOKIE_ALLOW

persistentCookie

Use persistent SSO cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

Possible values: ON, OFF

Default value: OFF

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistence cookie setting is enabled.

Minimum value: 1

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

Default value: "None"

Resets the attributes of the specified traffic session parameters. Attributes for which a default value is available revert to their default values. Refer to the set tm sessionParameter command for descriptions of the parameters..Refer to the set tm sessionParameter command for meanings of the arguments.

unset tm sessionParameter [-sessTimeout] [-SSO] [-ssoDomain] [-kcdAccount] [-persistentCookie] [-homePage] [-defaultAuthorizationAction] [-ssoCredential] [-httpOnlyCookie] [-persistentCookieValidity]

Displays information about traffic session parameters.

show tm sessionParameter

format

level

name

sessTimeout

The session timeout, in minutes.

defaultAuthorizationAction

The Authentication Action, e.g. allow or deny.

SSO

Whether or not Single Sign-On is used for this session.

ssoCredential

Use primary or secondary authentication credentials for single sign-on.

ssoDomain

Domain to use for single sign-on.

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

persistentCookie

Use persistent SSO cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistence cookie setting is enabled.

tm sessionPolicy

Sep 22, 2015

The following operations can be performed on "tm sessionPolicy":

add | rm | set | unset | show

Creates a traffic management (TM) session policy, which is applied after the user logs on to the AAA virtual server, to customize user sessions.

add tm sessionPolicy <name> <rule> <action>

name

Name for the session policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after a session policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied to connections that match this policy.

Removes an existing traffic management (TM) session policy.

rm tm sessionPolicy <name>

name

Name of the session policy to remove.

Modifies the rule or action of an existing traffic management (TM) session policy.

set tm sessionPolicy <name> [-rule <expression>] [-action <string>]

name

Name of the session policy to modify.

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied to connections that match this policy.

Use this command to remove tm sessionPolicy settings.Refer to the set tm sessionPolicy command for meanings of the arguments.

unset tm sessionPolicy <name> [-rule] [-action]

Displays information about all the configured traffic management (TM) session policies, or displays detailed information about the specified TM session policy.

show tm sessionPolicy [<name>]

name

Name of the session policy for which to display detailed information.

summary

fullValues

format

level

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied to connections that match this policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

tm trafficAction

Sep 22, 2015

The following operations can be performed on "tm trafficAction":

add | rm | set | unset | show

Creates a traffic action to set traffic characteristics at run time. You can create a traffic action for an application that is installed in the internal network (for example, an action that defines the destination IP address and destination port, and sets the amount of time a user can stay logged on to the application, such as 15 minutes).

add tm trafficAction <name> [-appTimeout <mins>] [-SSO (ON | OFF) [-formSSOAction <string>]] [-persistentCookie (ON | OFF)] [-InitiateLogout (ON | OFF)] [-kcdAccount <string>] [-samlSSOProfile <string>]

name

Name for the traffic action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after a traffic action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

appTimeout

Time interval, in minutes, of user inactivity after which the connection is closed.

Minimum value: 1

Maximum value: 715827

SSO

Use single sign-on for the resource that the user is accessing now.

Possible values: ON, OFF

formSSOAction

Name of the configured form-based single sign-on profile.

persistentCookie

Use persistent cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

Possible values: ON, OFF

InitiateLogout

Initiate logout for the traffic management (TM) session if the policy evaluates to true. The session is then terminated after two minutes.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "None"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

---

Removes an existing traffic action.

rm tm trafficAction <name>

name

Name of the traffic action to remove.

Modifies the specified parameters of an existing traffic action.

set tm trafficAction <name> [-appTimeout <mins>] [-SSO (ON | OFF)] [-formSSOAction <string>] [-persistentCookie (ON | OFF)] [-InitiateLogout (ON | OFF)] [-kcdAccount <string>] [-samlSSOProfile <string>]

name

Name of the traffic action to modify.

appTimeout

Time interval, in minutes, of user inactivity after which the connection is closed.

Minimum value: 1

Maximum value: 715827

SSO

Use single sign-on for the resource that the user is accessing now.

Possible values: ON, OFF

formSSOAction

Name of the configured form-based single sign-on profile.

persistentCookie

Use persistent cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

Possible values: ON, OFF

InitiateLogout

Initiate logout for the traffic management (TM) session if the policy evaluates to true. The session is then terminated after two minutes.

Possible values: ON, OFF

kcdAccount

Kerberos contrained delegation account name

Default value: "None"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

---

Use this command to remove tm trafficAction settings.Refer to the set tm trafficAction command for meanings of the arguments.

unset tm trafficAction <name> [-persistentCookie] [-kcdAccount]

Displays information about all configured traffic management (TM) traffic actions, or displays detailed information about the specified TM traffic action.

show tm trafficAction [<name>]

name

Name of the traffic action for which to display detailed information.

summary

fullValues

format

level

appTimeout

The application timeout

SSO

Whether or not Single Sign On is enabled.

formSSOAction

Name of the configured form-based single sign-on profile.

stateflag

persistentCookie

Use persistent cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

InitiateLogout

Whether Logout is initiated with this action

kcdAccount

Kerberos contrained delegation account name

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

devno

count

tm trafficPolicy

Sep 22, 2015

The following operations can be performed on "tm trafficPolicy":

add | rm | set | unset | show | stat

Adds a traffic policy to use for setting connection timeout, single sign-on, and initiating logout. The policy sets the characteristics of application traffic at run time.

add tm trafficPolicy <name> <rule> <action>

name

Name for the traffic policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to apply to requests or connections that match this policy.

Removes an existing traffic policy.

rm tm trafficPolicy <name>

name

Name of the traffic policy to remove.

Modifies the specified parameters of an existing traffic policy.

set tm trafficPolicy <name> [-rule <expression>] [-action <string>]

name

Name of the traffic policy to modify.

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to apply to requests or connections that match this policy.

Use this command to remove tm trafficPolicy settings.Refer to the set tm trafficPolicy command for meanings of the arguments.

unset tm trafficPolicy <name> [-rule] [-action]

Displays information about all configured traffic management (TM) traffic policies, or displays detailed information about the specified TM traffic policy.

show tm trafficPolicy [<name>]

name

Name of the traffic policy for which to display detailed information.

summary

fullValues

format

level

rule

The rule used by the vpn traffic policy. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

action

The action to be performed when the rule is matched.

stateflag

boundTo

The entity name to which policy is bound

activePolicy

priority

hits

Number of hits.

bindPolicyType

vserverType

devno

count

Display Traffic Management traffic policy statistics.

stat tm trafficPolicy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

The name of the TM traffic policy for which statistics will be displayed. If not given statistics are shown for all policies.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy hits (Hits)

Number of hits on the policy

stat tm trafficpolicy.

Transform Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- transform action
- transform global
- transform policy
- transform policylabel
- transform profile

transform action

Sep 22, 2015

The following operations can be performed on "transform action":

add | rm | set | unset | show

Creates a URL Transformation action, which defines how a specific element in URLs in the request or response is to be modified. NOTE: In the URL Transformation feature (unlike all other NetScaler features), ?profile? and ?action? are not synonymous but refer to distinct entities. You must create the profile first, and then the actions.

add transform action <name> <profileName> <priority> [-state (ENABLED | DISABLED)]

name

Name for the URL transformation action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the URL Transformation action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform action? or ?my transform action).

profileName

Name of the URL Transformation profile with which to associate this action.

priority

Positive integer specifying the priority of the action within the profile. A lower number specifies a higher priority. Must be unique within the list of actions bound to the profile. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

Minimum value: 1

Maximum value: 2147483647

state

Enable or disable this action.

Possible values: ENABLED, DISABLED

Default value: GENENABLED

Removes a URL Transformation action.

rm transform action <name>

name

Name of the action.

Modifies the settings of the specified URL Transformation action.

set transform action <name> [-priority <positive_integer>] [-reqUrlFrom <expression>] [-reqUrlInto <expression>] [-resUrlFrom <expression>] [-resUrlInto <expression>] [-cookieDomainFrom <expression>] [-cookieDomainInto <expression>] [-state (ENABLED | DISABLED)] [-comment <string>]

name

Name of the URL Transformation action to modify.

priority

Positive integer specifying the priority of the action within the profile. A lower number specifies a higher priority. Must be unique within the list of actions bound to the profile. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

Minimum value: 1

Maximum value: 2147483647

reqUrlFrom

PCRE-format regular expression that describes the request URL pattern to be transformed.

reqUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the reqUrlFrom pattern.

resUrlFrom

PCRE-format regular expression that describes the response URL pattern to be transformed.

resUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the resUrlFrom pattern.

cookieDomainFrom

Pattern that matches the domain to be transformed in Set-Cookie headers.

cookieDomainInto

PCRE-format regular expression that describes the transformation to be performed on cookie domains that match the cookieDomainFrom pattern.

NOTE: The cookie domain to be transformed is extracted from the request.

state

Enable or disable this action.

Possible values: ENABLED, DISABLED

Default value: GENENABLED

comment

Any comments to preserve information about this URL Transformation action.

Use this command to remove transform action settings.Refer to the set transform action command for meanings of the arguments.

unset transform action <name> [-reqUrlFrom] [-reqUrlInto] [-resUrlFrom] [-resUrlInto] [-cookieDomainFrom] [-cookieDomainInto] [-state] [-comment]

Displays a list of all URL Transformation actions currently assigned to the specified profile.

show transform action [<name>]

name

Name of the profile.

summary

fullValues

format

level

stateflag

profileName

Name of the URL Transformation profile with which to associate this action.

priority

Positive integer specifying the priority of the action within the profile. A lower number specifies a higher priority. Must be unique within the list of actions bound to the profile. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

reqUrlFrom

PCRE-format regular expression that describes the request URL pattern to be transformed.

reqUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the reqUrlFrom pattern.

resUrlFrom

PCRE-format regular expression that describes the response URL pattern to be transformed.

resUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the resUrlFrom pattern.

cookieDomainFrom

Pattern that matches the domain to be transformed in Set-Cookie headers.

cookieDomainInto

PCRE-format regular expression that describes the transformation to be performed on cookie domains that match the cookieDomainFrom pattern.

NOTE: The cookie domain to be transformed is extracted from the request.

continueMatching

Continue transforming using the next rule in the list.

state

Enable or disable this action.

comment

Any comments to preserve information about this URL Transformation action.

devno

count

transform global

Sep 22, 2015

The following operations can be performed on "transform global":

bind | unbind | show

Activates the specified URL Transformation policy for all traffic received by this NetScaler appliance. If you set policyName to a name that does not match an existing URL Transformation policy name, this command creates the policy, with the configuration that you specify.

bind transform global <policyName> <priority> [<gotoPriorityExpression>] [-type (REQ_OVERRIDE | REQ_DEFAULT)] [-invoke (<labelType> <labelName>)]

policyName

Name of the policy.

If you want to create the policy as well as activate it, specify a name for the policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policy? or ?my transform policy).

bind transform global pol9 9

Unbinds the specified URL Transformation policy from URL Transformation global.

unbind transform global <policyName> [-type (REQ_OVERRIDE | REQ_DEFAULT)] [-priority <positive_integer>]

policyName

The name of the policy to be unbound.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

unbind transform global pol9

Displays the policies bound to the specified URL Transformation global bind point. If no bind point is specified, displays a list of all policies bound to URL Transformation global.

show transform global [-type (REQ_OVERRIDE | REQ_DEFAULT)]

type

Specifies the bind point to which to bind the policy. Available settings function as follows:

* REQ_OVERRIDE. Request override. Binds the policy to the priority request queue.

* REQ_DEFAULT. Binds the policy to the default request queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT

summary

fullValues

format

level

stateflag

policyName

Name of the transform policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forwards the request or response to the specified virtual server or evaluates the specified policy label.

labelType

Type of invocation. Available settings function as follows:

* reqvserver - Send the request to the specified request virtual server.

* resvserver - Send the response to the specified response virtual server.

* policylabel - Invoke the specified policy label.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and the label type is Policy Label.

flowType

flowtype of the bound transform policy.

numpol

The number of policies bound to the bindpoint.

flags

devno

count

show transform global

transform policy

Sep 22, 2015

The following operations can be performed on "transform policy":

add | rm | set | unset | show | stat | rename

Creates a URL Transformation policy, which specifies the requests and responses to be transformed by the associated profile.

add transform policy <name> <rule> <profileName> [-comment <string>] [-logAction <string>]

name

Name for the URL Transformation policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the URL Transformation policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policy? or ?my transform policy).

rule

Expression, or name of a named expression, against which to evaluate traffic. Can be written in either default or classic syntax. Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the URL Transformation profile to use to transform requests and responses that match the policy.

comment

Any comments to preserve information about this URL Transformation policy.

logAction

Log server to use to log connections that match this policy.

Removes the specified URL Transformation policy.

rm transform policy <name>

name

Name of the policy to remove.

rm transform policy trans_pol

Modifies the specified parameters of a URL Transformation policy.

set transform policy <name> [-rule <expression>] [-profileName <string>] [-comment <string>] [-logAction <string>]

name

Name of the policy to modify.

rule

Expression, or name of a named expression, against which to evaluate traffic. Can be written in either default or classic syntax. Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the URL Transformation profile to use to transform requests and responses that match the policy.

comment

Any comments to preserve information about this URL Transformation policy.

logAction

Log server to use to log connections that match this policy.

```
set transform policy pol9 -rule "HTTP.REQ.HEADER(\\\"header\\\").CONTAINS(\\\"qh2\\\")"
```

Removes the settings of an existing URL Transformation policy. Attributes for which a default value is available revert to their default values. See the set transform policy command for a description of the parameters..Refer to the set transform policy command for meanings of the arguments.

```
unset transform policy <name> [-comment] [-logAction]
```

```
unset transform policy pol9 -undefAction
```

Displays the current settings for the specified URL Transformation policy. If no policy name is specified, displays a list of all URL Transformation policies currently configured on the NetScaler appliance.

```
show transform policy [<name>]
```

name

Name of the URL Transformation policy.

summary

fullValues

format

level

stateflag

rule

Expression, or name of a named expression, against which to evaluate traffic. Can be written in either default or classic syntax. Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the URL Transformation profile to use to transform requests and responses that match the policy.

priority

Specifies the priority of the policy.

hits

Number of hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any comments to preserve information about this URL Transformation policy.

logAction

Log server to use to log connections that match this policy.

bindPolicyType

isDefault

A value of true is returned if it is a default transform policy.

vserverType

devno

count

Displays statistics for the specified URL Transformation policy. If no policy name is specified, displays abbreviated statistics for all URL Transformation policies currently configured on the NetScaler appliance.

stat transform policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

name

Name of the policy.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

stat transform policy

Renames a URL Transformation policy.

rename transform policy <name>@
<newName>@

name

Existing name of the policy.

newName

New name for the policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policy? or ?my transform policy).

rename transform policy oldname newname

transform policylabel

Sep 22, 2015

The following operations can be performed on "transform policylabel":

add | rm | bind | unbind | show | stat | rename

Creates a URL Transformation policy label. A policy label is a tool for evaluating a set of policies in a specified order. By using a policy label, you can configure the URL Transformation feature to choose the next policy, invoke a different policy label, or terminate policy evaluation completely by looking at whether the previous policy evaluated to TRUE or FALSE.

add transform policylabel <labelName> <policylabeltype>

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the URL Transformation policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policylabel? or ?my transform policylabel).

policylabeltype

Types of transformations allowed by the policies bound to the label. For URL transformation, always http_req (HTTP Request).

Possible values: http_req

```
add transform policylabel trans_policylabel http_req
```

Removes a URL Transformation policy label.

rm transform policylabel <labelName>

labelName

Name of the policy label to remove.

```
rm transform policylabel trans_policylabel
```

Binds the specified URL Transformation policy to the specified policy label.

bind transform policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

labelName

Name of the URL Transformation policy label to which to bind the policy.

policyName

Name of the URL Transformation policy to bind to the policy label.

```
i) bind transform policylabel trans_policylabel pol_1 1 2 -invoke reqvserver CURRENT  ii) bind transform policylabel trans_policylabel pol_2 2
```

Unbinds the specified URL Transformation policy from the specified policy label.

```
unbind transform policylabel <labelName> <policyName> [-priority <positive_integer>]
```

labelName

Name of the label from which to unbind the policy.

policyName

Name of the label to which to bind the policy.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

```
unbind transform policylabel trans_policylabel pol_1
```

Displays the current settings for the specified URL Transformation policy label. If no policy label is specified, displays a list of all URL Transformation policy labels currently configured on the NetScaler appliance.

```
show transform policylabel [<labelName>]
```

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Can be changed after the URL Transformation policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policylabel? or ?my transform policylabel).

summary

fullValues

format

level

stateflag

policylabeltype

Types of transformations allowed by the policies bound to the label. For URL transformation, always http_req (HTTP Request).

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the URL Transformation policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

* reqvserver - Forward the request to the specified request virtual server.

* policylabel - Invoke the specified policy label.

labelName

Name of the policy label.

description

Description of the policylabel

flags

devno

count

i) show transform policylabel trans_policylabel ii) show transform policylabel

Displays statistics for the specified URL Transformation policy label. If no policy label name is provided, displays abbreviated statistics for all URL Transformation policy labels currently configured on the NetScaler appliance.

stat transform policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

labelName

The name of the URL Transformation policy label.

clearstats

Clear the statsistics / counters

Possible values: basic, full

count

devno

stateflag

Policy Label Hits (Hits)

Number of times policy label was invoked.

Renames a URL Transformation policy label.

rename transform policylabel <labelName>@ <newName>@

labelName

Current name of the policy label.

newName

New name for the policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policylabel? or ?my transform policylabel).

rename transform policylabel oldname newname

---

transform profile

Sep 22, 2015

The following operations can be performed on "transform profile":

add | rm | set | unset | show

Creates a URL transformation profile, which contains a list of actions that define how the URLs in a request or response are to be modified. NOTE: In the URL Transformation feature (unlike all other NetScaler features), ?profile? and ?action? are not synonymous but refer to distinct entities. You must create the profile first, and then the actions.

add transform profile <name> [-type URL]

name

Name for the URL transformation profile. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the URL transformation profile is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform profile? or ?my transform profile?).

type

Type of transformation. Always URL for URL Transformation profiles.

Possible values: URL

Removes a URL Transformation profile.

rm transform profile <name>

name

Name of the profile to remove.

Modifies the settings of a URL Transformation profile.

```
set transform profile <name> [-type URL] [-onlyTransformAbsURLinBody ( ON | OFF )] [-comment <string>]
```

name

Name of the profile to be modified.

type

Type of transformation. Always URL for URL Transformation profiles.

Possible values: URL

onlyTransformAbsURLinBody

In the HTTP body, transform only absolute URLs. Relative URLs are ignored.

Possible values: ON, OFF

comment

Any comments to preserve information about this URL Transformation profile.

Use this command to remove transform profile settings.Refer to the set transform profile command for meanings of the arguments.

```
unset transform profile <name> [-type] [-onlyTransformAbsURLinBody] [-comment]
```

Displays the current settings for the specified URL Transformation profile. If no URL Transformation profile name is specified, displays a list of all URL Transformation profiles currently configured on the NetScaler appliance.

```
show transform profile [<name>]
```

name

Name of the profile.

summary

fullValues

format

level

actionName

URL Transformation action name.

stateflag

type

Type of transformation. Always URL for URL Transformation profiles.

RegexForFindingURLinJavaScript

Patclass having regexes to find the URLs in JavaScript.

RegexForFindingURLinCSS

Patclass having regexes to find the URLs in CSS.

RegexForFindingURLinXComponent

Patclass having regexes to find the URLs in X-Component.

RegexForFindingURLinXML

Patclass having regexes to find the URLs in XML.

additionalReqHeadersList

Patclass having a list of additional request header names that should transformed.

additionalRespHeadersList

Patclass having a list of additional response header names that should transformed.

onlyTransformAbsURLinBody

In the HTTP body, transform only absolute URLs. Relative URLs are ignored.

comment

Any comments to preserve information about this URL Transformation profile.

priority

Priority of the Action within the Profile.

state

Enabled flag.

profileName

URL Transformation profile name.

reqUrlFrom

Pattern of original request URLs. It corresponds to the way external users view the server, and acts as a source for request transformations.

reqUrlInto

Pattern of transformed request URLs. It corresponds to internal addresses and indicates how they are created.

resUrlFrom

Pattern of original response URLs. It corresponds to the way external users view the server, and acts as a source for response transformations.

resUrlInto

Pattern of transformed response URLs. It corresponds to internal addresses and indicates how they are created.

cookieDomainFrom

Pattern of the original domain in Set-Cookie headers.

cookieDomainInto

Pattern of the transformed domain in Set-Cookie headers.

actionComment

Comments.

devno

count

Tunnel Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- tunnel global
- tunnel trafficPolicy

tunnel global

Sep 22, 2015

The following operations can be performed on "tunnel global":

bind | unbind | show

Activates an existing tunnel traffic policy globally.

bind tunnel global (<policyName> [-priority <positive_integer>]) [-state (ENABLED | DISABLED)]

policyName

Name of the tunnel traffic policy to activate or bind.

add tunnel trafficpolicy cmp_all_destport "REQ.TCP.DESTPORT == 0-65535" GZIP After creating above tunnel policy, it can be activated by binding it globally: bind tun

---

Deactivates an active tunnel traffic policy.

unbind tunnel global <policyName>

policyName

Name of the tunnel traffic policy to unbind or deactivate.

Globally active tunnel policies can be seen using command: > show tunnel global 1 Globally Active Tunnel Policies: 1) Policy Name: cmp_all_destport Priority: 0 Don

---

Displays globally active tunnel policies.

show tunnel global

summary

fullValues

format

level

policyName

Policy name.

priority

Priority.

state

Current state of the binding. If the binding is enabled, the policy is active.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

```
> sh tunnel global 1) Policy Name: cmp_all_destport   Priority: 0 2) Policy Name: local_sub_nocmp   Priority: 500  Done
```

tunnel trafficPolicy

Sep 22, 2015

The following operations can be performed on "tunnel trafficPolicy":

add | rm | set | unset | show

Creates a tunnel traffic policy. A tunnel traffic policy defines the type of compression to be used for the tunneled traffic.

add tunnel trafficPolicy <name> <rule> <action>

name

Name for the tunnel traffic policy.

Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy)'.

rule

Expression, against which traffic is evaluated. Written in classic or default syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in compression action to associate with the policy.

Example 1: add tunnel trafficpolicy cmp_all_destport "REQ.TCP.DESTPORT == 0-65535" GZIP After creating above tunnel policy, it can be activated by binding it global

---

Removes a tunnel traffic policy.

rm tunnel trafficPolicy <name>

name

Name of the tunnel traffic policy to remove.

rm tunnel trafficpolicy tunnel_policy_name The "show tunnel trafficpolicy" command shows all tunnel policies that are currently defined.

Modifies the specified parameters of an existing tunnel traffic policy.

set tunnel trafficPolicy <name> [-rule <expression>] [-action <string>]

name

Name of the tunnel traffic policy to modify.

rule

Expression, against which traffic is evaluated. Written in classic or default syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in compression action to associate with the policy.

```
add tunnel trafficpolicy cmp_all_destport "REQ.TCP.DESTPORT == 0-65535" GZIP  set tunnel trafficpolicy cmp_all_destport -action NOCOMPRESS  Above 'set' comman
```

Use this command to remove tunnel trafficPolicy settings.Refer to the set tunnel trafficPolicy command for meanings of the arguments.

```
unset tunnel trafficPolicy <name> [-rule] [-action]
```

Displays information about all the configured tunnel traffic policies, or displays detailed information about the specified tunnel traffic policy.

```
show tunnel trafficPolicy [<name>]
```

name

Name of the tunnel traffic policy for which to show detailed information.

summary

fullValues

format

level

rule

Expression, against which traffic is evaluated. Written in classic or default syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

* If the expression itself includes double quotation marks, you must escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in compression action to associate with the policy.

hits

No of hits.

txbytes

Number of bytes transmitted.

rxbytes

Number of bytes received.

clientTTLB

Total client TTLB value.

clientTransactions

Number of client transactions.

serverTTLB

Total server TTLB value.

serverTransactions

Number of server transactions.

boundTo

The entity name to which policy is bound

activePolicy

priority

flags

bindPolicyType

isDefault

A value of true is returned if it is a default tunnelpolicy.

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

> show tunnel trafficpolicy 2 Tunnel policies: 1) Name: local_sub_nocmp Rule: SOURCEIP =

Utility Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- audit
- callhome
- grep
- install
- nstrace
- ping
- ping6
- raid
- scp
- shell
- techsupport
- traceroute
- traceroute6

audit

Sep 22, 2015

The following operations can be performed on "audit":

audit and verify the commands in file against running config. NOTE: This command is deprecated.Command deprecated. Use diff ns config command

commandStr

specify the options.

```
config audit -diff -f <filename>
```

callhome

Sep 22, 2015

The following operations can be performed on "callhome":

show | set | unset

Displays the trigger events configured and the time when these events were triggered.

show callhome

format

level

emailAddress

The contact person's E-mail address.

proxyMode

Deploy the callhome proxy mode

IPAddress

Proxy Server IP address

port

Proxy Server Port

sslcardfirstfailure

First occurrence SSL card failure.

sslcardlatestfailure

Latest occurrence SSL card failure.

powfirstfail

First occurrence power supply unit failure.

powlatestfailure

Latest occurrence power supply unit failure.

HDDfirstfail

First occurrence hard disk drive failure.

HDDlatestfailure

Latest occurrence hard disk drive failure.

FLASHfirstfail

First occurrence compact flash failure.

FLASHlatestfailure

Latest occurrence compact flush failure.

restartLatestfail

Latest occurrence warm restart failure.

callhomestatus

Callhome feature enabled/disable, register with upload server successful/failed

show callhome E-mail address configured:xxx@yahoo.com Trigger event State First occurrence Latest occurrence ---------- ----- -----

Sets the contact person's E-mail address

set callhome -emailAddress e-mailaddress

emailAddress

The contact person's E-mail address.

proxyMode

Deploy the callhome proxy mode

Possible values: YES, NO

Default value: NO

```
set callhome -emailAddress xxxx@yahoo.com
```

Use this command to remove callhome settings.Refer to the set callhome command for meanings of the arguments.

```
unset callhome [-emailAddress] [-proxyMode] [-IPAddress] [-port]
```

---

grep

Sep 22, 2015

The following operations can be performed on "grep":

Searches files or output for lines containing a match to the specified <pattern>. By default, grep prints the matching lines.

grep [-c] [-E] [-i] [-v] [-w] [-x] <pattern>

c

Suppress normal output. Instead print a count of matching lines.

With the -v option, count non-matching lines.

E

Interpret <pattern> as an extended regular expression.

i

Ignore case distinctions.

v

Invert the sense of matching, to select non-matching lines.

w

Select only those lines containing matches that form whole words.

x

Select only those matches that exactly match the whole line.

pattern

The pattern (regular expression or text string) for which to search.

```
show ns info | grep off -i
```

install

Sep 22, 2015

The following operations can be performed on "install":

Installs a version of NetScaler software on the system.

install <url> [-c] [-y]

url

http://[user]:[password]@host/path/to/file

https://[user]:[password]@host/path/to/file

sftp://[user]:[password]@host/path/to/file

scp://[user]:[password]@host/path/to/file

ftp://[user]:[password]@host/path/to/file

file://path/to/file

c

Back up existing kernel.

y

Do not prompt for yes/no before rebooting.

```
install http://host.netscaler.com/ns-6.0-41.2.tgz
```

nstrace

Sep 22, 2015

The following operations can be performed on "nstrace":

Invokes the nstrace program to log traffic flowing through the NetScaler appliance.

nstrace [-nf <positive_integer>] [-time <secs>] [-size <positive_integer>] [-mode <mode> ...] [-tcpdump (ENABLED | DISABLED) [-perNIC (ENABLED | DISABLED)]] [-name <string> [-id <string>]] [-filter <expression> [-link (ENABLED | DISABLED)]]

h

prints this message - exclusive option

nf

Number of files to be generated in a single run of the command.

Default value: 24

time

Number of seconds for which to log to trace file. Can be a mathematical expression. For example, to log to trace files for 2 hours, you can specify 2*60*60.

Default value: 3600

size

Size of the packet to be logged (should be in the range of 60 to 1514 bytes). Set to 0 for full packet trace.

Default value: 164

Maximum value: 1514

m

Capturing mode: sum of the values:

1 - Transmitted packets (TX)

2 - Packets buffered for transmission (TXB)

4 - Received packets (RX)

Default value: 6

tcpDump

Log files in TCP dump format (instead of nstrace format).

Possible values: NSTRACE, TCPDUMP

mode

Capturing mode for trace. Can be any of the following values, or a combination of these values:

* RX - Received packets before NIC pipelining

* NEW_RX - Received packets after NIC pipelining (packets that are not dropped)

* TX - Transmitted packets

* TXB - Packets buffered for transmission

* IPV6 - Translated IPv6 packets

* C2C - Capture core-to-core messages

* NS_FR_TX - Flow receiver does not capture the TX/TXB packets. Applicable only for a cluster setup.

You can also provide a combination of modes. For example:

* -mode NEW_RX TXB: Capture RX packets after NIC handling and packets that are buffered for actual transmission.

* -mode RX TX: Capture packet during NIC pipeline (filter expressions will not work for RX mode).

* -mode NEW_RX TXB NS_FR_TX: Default mode except that TX/TXB packets on the flow receiver are not captured.

Default value: DEFAULT_MODE

tcpdump

Log files format supported:nstrace-format, tcpdump-format. default:nstrace-format

Possible values: ENABLED, DISABLED

Default value: DISABLED

name

Custom file name for nstrace files.

filter

Filter expression for nstrace. Maximum length of filter is 255 and it can be of the following format:

"<expression> [<relop> <expression>"]

where,

<relop> can be the && or the || relational operators.

<expression> is a string in the following format: <qualifier> <operator> <qualifier-value>

where,

<operator> can be any one of the following (except the commas): ==, eq, !=, neq, >, gt, <, lt, >=, ge, <=, le, BETWEEN

Following are the valid qualifiers for the command: SOURCEIP, SOURCEPORT, DESTIP, DESTPORT, IP, PORT, SVCNAME, VSVRNAME, CONNID, VLAN, INTF.

Example:

nstrace -filter "SOURCEIP==10.102.34.201 || SVCNAME !=s1 && SOURCEPORT >80"

```
nstrace -nf 10 -time 100 -mode RX IPV6 TXB -name abc -tcpdump ENABLED -perNIC ENABLED
```

ping

Sep 22, 2015

The following operations can be performed on "ping":

Invokes the UNIX ping command. The hostName parameter must be used if the name is in the /etc/hosts file directory or is otherwise known in DNS.

ping [-c <count>] [-i <interval>] [-I <interface>] [-n] [-p <pattern>] [-q] [-s <size>] [-S <src_addr>] [-T <td>] [-t <timeout>] <hostname>

c

Number of packets to send. The default value is infinite.

Minimum value: 1

Maximum value: 65535

i

Waiting time, in seconds. The default value is 1 second.

Maximum value: 65535

I

Network interface on which to ping, if you have multiple interfaces.

n

Numeric output only. No name resolution.

p

Pattern to fill in packets. Can be up to 16 bytes, useful for diagnosing data-dependent problems.

q

Quiet output. Only the summary is printed.

s

Data size, in bytes. The default value is 56.

Maximum value: 65507

S

Source IP address to be used in the outgoing query packets. If the IP addrESS does not belongs to this appliance, an error is returned and nothing is sent.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

t

Time-out, in seconds, before ping exits.

Minimum value: 1

Maximum value: 3600

hostName

Address of host to ping.

```
ping -p ff -c 4 10.102.4.107
```

ping6

Sep 22, 2015

The following operations can be performed on "ping6":

Invokes the UNIX ping6 command. The hostName parameter must be used if the name is in the /etc/hosts file directory or is otherwise known in DNS.

ping6 [-c <count>] [-i <interval>] [-I <interface>] [-n] [-p <pattern>] [-q] [-S sourceaddr] [-V <vlanid>] [-T <td>] [-s <size>] Hostname

c

Number of packets to send. The default value is infinite.

Minimum value: 1

Maximum value: 65535

i

Waiting time, in seconds. The default value is 1 second.

Maximum value: 65535

I

Network interface on which to ping, if you have multiple interfaces.

n

Numeric output only. No name resolution.

p

Pattern to fill in packets. Can be up to 16 bytes, useful for diagnosing data-dependent problems.

q

Quiet output. Only summary is printed.

s

Data size, in bytes. The default value is 56.

Maximum value: 65507

V

VLAN ID for link local address.

Minimum value: 1

Maximum value: 4094

S

Source IP address to be used in the outgoing query packets.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

t

Timeout in seconds before ping6 exits

hostName

Address of host to ping.

```
ping6 -p ff -I 1/1 -c 4 2002::1
```

raid

Sep 22, 2015

The following operations can be performed on "raid":

Provides status of raid

show raid

status raid

scp

Sep 22, 2015

The following operations can be performed on "scp":

Securely copies data from one computer to another, in SSH protocol.

scp [-r] [-C] [-q] <sourceString> <destString>

r

Recursively copy subdirectories.

C

Enable compression.

q

Quiet output. Disable the progress meter.

sourceString

Source user, host, and file path, specified as <user>@<host>:<path_to_copy_from>. The user and host parts are optional.

destString

Destination user, host, and file path, specified as

<user>@<host>:<path_to_copy_to>. The user and host parts are optional.

```
scp /nsconfig/ns.conf nsroot@10.102.4.107:/nsconfig/
```

shell

Sep 22, 2015

The following operations can be performed on "shell":

Exits to the FreeBSD command prompt. Press Control + D or type exit to return to the NetScaler command prompt. Note: The shell can be accessed only by users who have write access to the NetScaler appliance.

shell [(command)]

command

Shell command(s) to be invoked.

techsupport

Sep 22, 2015

The following operations can be performed on "techsupport":

Generates a tar of system configuration data and statistics. This file must be submitted to Citrix technical support with file name collector_<NS IP>_<P/S>_<DateTime>.tgz. The archive is always pointed by the symbolic link /var/tmp/support/support.tgz for each invocation of the command.

show techsupport [-scope (NODE | CLUSTER)]

scope

Use this option to run showtechsupport on present node or all cluster nodes.

Possible values: NODE, CLUSTER

Default value: NS_TECH_NODE

serverName

show techsupport

traceroute

Sep 22, 2015

The following operations can be performed on "traceroute":

Invokes the UNIX traceroute command. This command attempts to track the route that the packets follow to reach the destination host.

traceroute [-S] [-n] [-r] [-v] [-M <min_ttl] [-m <max_ttl>] [-P <protocol>][-p <portno>] [-q <nqueries>] [-s <src_addr>] [-T <td>] [-t <tos>] [-w <wait>] <host> [<packetlen>]

S

Print a summary of how many probes were not answered for each hop.

n

Print hop addresses numerically instead of symbolically and numerically.

r

Bypass normal routing tables and send directly to a host on an attached network. If the host is not on a directly attached network, an error is returned.

v

Verbose output. List received ICMP packets other than TIME_EXCEEDED and UNREACHABLE.

M

Minimum TTL value used in outgoing probe packets.

Default value: 1

Minimum value: 1

Maximum value: 255

m

Maximum TTL value used in outgoing probe packets.

Default value: 64

Minimum value: 1

Maximum value: 255

P

Send packets of specified IP protocol. The currently supported protocols are UDP and ICMP.

p

Base port number used in probes.

Default value: 33434

Minimum value: 1

Maximum value: 65535

q

Number of queries per hop.

Default value: 3

Minimum value: 1

Maximum value: 65535

s

Source IP address to use in the outgoing query packets. If the IP address does not belong to this appliance, an error is returned and nothing is sent.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

t

Type-of-service in query packets.

Maximum value: 255

w

Time (in seconds) to wait for a response to a query.

Default value: 5

Minimum value: 2

Maximum value: 86399

host

Destination host IP address or name.

packetlen

Length (in bytes) of the query packets.

Default value: 44

Minimum value: 44

Maximum value: 32768

traceroute 10.102.4.107

traceroute6

Sep 22, 2015

The following operations can be performed on "traceroute6":

Invokes the UNIX traceroute6 command. Traceroute6 attempts to track the route that the packets follow to reach the destination host.

traceroute6 [-n] [I] [-r] [-v] [-m <hoplimit>] [-p <port>] [-q <probes>] [-s <src_addr>] [-T <td>] [-w <waittime>] <target> [<packetlen>]

n

Print hop addresses numerically rather than symbolically and numerically.

I

Use ICMP ECHO for probes.

r

Bypass normal routing tables and send directly to a host on an attached network. If the host is not on a directly attached network, an error is returned.

v

Verbose output. List received ICMP packets other than TIME_EXCEEDED and UNREACHABLE.

m

Maximum hop value for outgoing probe packets.

Default value: 64

Minimum value: 1

Maximum value: 255

p

Base port number used in probes.

Default value: 33434

Minimum value: 1

Maximum value: 65535

q

Number of probes per hop.

Default value: 3

Minimum value: 1

Maximum value: 65535

s

Source IP address to use in the outgoing query packets. If the IP address does not belong to this appliance, an error is returned and nothing is sent.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

w

Time (in seconds) to wait for a response to a query.

Default value: 5

Minimum value: 2

Maximum value: 86399

host

Destination host IP address or name.

packetlen

Length (in bytes) of the query packets.

Default value: 44

Minimum value: 44

Maximum value: 32768

```
traceroute6 2002::7
```

VPN Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- vpn
- vpn clientlessAccessPolicy
- vpn clientlessAccessProfile
- vpn formSSOAction
- vpn global
- vpn icaConnection
- vpn intranetApplication
- vpn nextHopServer
- vpn parameter
- vpn samlSSOProfile
- vpn sessionAction
- vpn sessionPolicy
- vpn stats
- vpn trafficAction
- vpn trafficPolicy
- vpn url
- vpn vserver

vpn

Sep 22, 2015

The following operations can be performed on "vpn":

stat vpn

Displays the statistics for Access Gateway usage. Displays event information, such as the event that generated the message, a time stamp, the message type, and predefined log levels and message information.

Synopsys

stat vpn [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

Login-page requests received (iHtHit)

Number of requests for VPN login page.

Login-page delivery failures (iHtFail)

Number of failures to display VPN login page.

Client-configuration requests (cfgHit)

Number of client configuration requests received by VPN server.

DNS queries resolved (dnsHit)

Number of DNS queries resolved by VPN server.

WINS queries resolved (winsHit)

Number of WINS queries resolved by VPN server.

Number of SSLVPN tunnels (csHit)

Number of SSL VPN tunnels formed between VPN server and client.

Backend non-HTTP server probes (csNoHttp)

Number of probes from VPN to back-end non-HTTP servers that have been accessed by the VPN client.

Backend HTTP server probes (csHttp)

Number of probes from VPN to back-end HTTP servers that have been accessed by the VPN client.

Backend server probe successes (csConSuc)

Number of successful probes to all back-end servers.

File-system requests received (totFsHit)

Number of file system requests received by VPN server.

IIP disabled and MIP used (IIPdMIPu)

Number of times MIP is used as IIP is disabled.

IIP failed and MIP used (IIPfMIPu)

Number of times MIP is used as IIP assignment failed.

IIP spillover and MIP used (IIPsMIPu)

Number of times MIP is used on IIP Spillover.

IIP disabled and MIP disabled (IIPdMIPd)

Both IIP and MIP is disabled.

IIP failed and MIP disabled (IIPfMIPd)

Number of times IIP assignment failed and MIP is disabled.

SOCKS method request received (SOCKSmReqR)

Number of received SOCKS method request.

SOCKS method request sent (SOCKSmReqS)

Number of sent SOCKS method request.

SOCKS method response received (SOCKSmRespR)

Number of received SOCKS method response.

SOCKS method response sent (SOCKSmRespS)

Number of sent SOCKS method response.

SOCKS connect request received (SOCKScReqR)

Number of received SOCKS connect request.

SOCKS connect request sent (SOCKScReqS)

Number of sent SOCKS connect request.

SOCKS connect response received (SOCKScRespR)

Number of received SOCKS connect response.

SOCKS connect response sent (SOCKScRespS)

Number of sent SOCKS connect response.

SOCKS server error (SOCKSserverErr)

Number of SOCKS server error.

SOCKS client error (SOCKSclientErr)

Number of SOCKS client error.

STA connection success (STAconnSucc)

Number of STA connection success.

STA connection failure (STAconnFail)

Number of STA connection failure.

CPS connection success (CPSconnSucc)

Number of CPS connection success.

CPS connection failure (CPSconnFail)

Number of CPS connection failure.

STA request sent (STAreqSent)

Number of STA request sent.

STA response received (STArespRecvd)

Number of STA response received.

ICA license failure (ICAlicenseFail)

Number of ICA license failure.

vpn clientlessAccessPolicy

Sep 22, 2015

The following operations can be performed on "vpn clientlessAccessPolicy":

add | rm | set | show

add vpn clientlessAccessPolicy

Adds a clientless access policy, which enables users to log on using a web browser and connect to the bookmarked web address without requiring the user to install a software plug-in.

Synopsys

add vpn clientlessAccessPolicy <name> <rule> <profileName>

Arguments

name

Name of the new clientless access policy.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the profile to invoke for the clientless access.

---

rm vpn clientlessAccessPolicy

Removes a clientless access policy.

Synopsys

rm vpn clientlessAccessPolicy <name>

Arguments

name

Name of the clientless access policy to remove.

---

set vpn clientlessAccessPolicy

Adds a new rule to be used by an existing clientless access policy that includes a simple expression that specifies the conditions for which the policy is enforced.

Synopsys

set vpn clientlessAccessPolicy <name> [-rule <expression>] [-profileName <string>]

Arguments

name

Name of the existing clientless access policy to modify.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "'<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the profile to invoke for the clientless access.

---

show vpn clientlessAccessPolicy

Displays a clientless access policy.

Synopsys

show vpn clientlessAccessPolicy [<name>]

Arguments

name

Name of the clientless access policy to display.

summary

fullValues

format

level

Outputs

rule

The rule used by the clientless access policy. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

profileName

The profile to invoked for the clientless access.

undefAction

The UNDEF action.

hits

The number of times the policy evaluated to true.

undefHits

The number of times the policy evaluation resulted in undefined processing.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound.

priority

Specifies the priority of the policy.

description

Description of the clientless access policy.

isDefault

A value of true is returned if it is a default vpnclientlessrwpolicy.

stateflag

devno

count

vpn clientlessAccessProfile

Sep 22, 2015

The following operations can be performed on "vpn clientlessAccessProfile":

add | rm | set | unset | show

add vpn clientlessAccessProfile

Adds a collection of settings that allows clientless access to a given application. Settings include the policies to specify whether to rewrite a URL, rules to find the URLs within various web content-types, and a set of cookies that are required to be present on the client machine.

Synopsys

add vpn clientlessAccessProfile <profileName>

Arguments

profileName

Name for the Access Gateway clientless access profile. Must begin with an ASCII alphabetic or underscore (_) character, and must consist only of ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

---

rm vpn clientlessAccessProfile

Removes a clientless access profile.

Synopsys

rm vpn clientlessAccessProfile <profileName>

Arguments

profileName

Name of the clientless access profile to remove.

---

set vpn clientlessAccessProfile

Modifies the settings for an existing clientless access profile.

Synopsys

set vpn clientlessAccessProfile <profileName> [-URLRewritePolicyLabel <string>] [-JavaScriptRewritePolicyLabel <string>] [-ReqHdrRewritePolicyLabel <string>] [-ResHdrRewritePolicyLabel <string>] [-RegexForFindingURLinJavaScript <string>] [-RegexForFindingURLinCSS <string>] [-RegexForFindingURLinXComponent <string>] [-RegexForFindingURLinXML <string>] [-RegexForFindingCustomURLs <string>] [-ClientConsumedCookies <string>] [-requirePersistentCookie (ON | OFF)]

Arguments

profileName

Name of the clientless access profile to modify.

URLRewritePolicyLabel

Name of the configured URL rewrite policy label. If you do not specify a policy label name, then URLs are not rewritten.

JavaScriptRewritePolicyLabel

Name of the configured JavaScript rewrite policy label. If you do not specify a policy label name, then JAVA scripts are not rewritten.

ReqHdrRewritePolicyLabel

Name of the configured Request rewrite policy label. If you do not specify a policy label name, then requests are not rewritten.

ResHdrRewritePolicyLabel

Name of the configured Response rewrite policy label.

RegexForFindingURLinJavaScript

Name of the pattern set that contains the regular expressions, which match the URL in Java script.

RegexForFindingURLinCSS

Name of the pattern set that contains the regular expressions, which match the URL in the CSS.

RegexForFindingURLinXComponent

Name of the pattern set that contains the regular expressions, which match the URL in X Component.

RegexForFindingURLinXML

Name of the pattern set that contains the regular expressions, which match the URL in XML.

RegexForFindingCustomURLs

Name of the pattern set that contains the regular expressions, which match the URLs in the custom content type other than HTML, CSS, XML, XCOMP, and JavaScript. The custom content type should be included in the patset ns_cvpn_custom_content_types.

ClientConsumedCookies

Specify the name of the pattern set containing the names of the cookies, which are allowed between the client

and the server. If a pattern set is not specified, Access Gateway does not allow any cookies between the client and the server. A cookie that is not specified in the pattern set is handled by Access Gateway on behalf of the client.

requirePersistentCookie

Specify whether a persistent session cookie is set and accepted for clientless access. If this parameter is set to ON, COM objects, such as MSOffice, which are invoked by the browser can access the files using clientless access. Use caution because the persistent cookie is stored on the disk.

Possible values: ON, OFF

Default value: OFF

---

unset vpn clientlessAccessProfile

Resets the attributes of the specified clientless access profile. Attributes for which a default value is available revert to their default values. Refer to the set vpn clientlessAccessProfile command for a description of the parameters..Refer to the set vpn clientlessAccessProfile command for meanings of the arguments.

Synopsys

unset vpn clientlessAccessProfile <profileName> [-URLRewritePolicyLabel] [-JavaScriptRewritePolicyLabel] [-ReqHdrRewritePolicyLabel] [-ResHdrRewritePolicyLabel] [-RegexForFindingURLinJavaScript] [-RegexForFindingURLinCSS] [-RegexForFindingURLinXComponent] [-RegexForFindingURLinXML] [-RegexForFindingCustomURLs] [-ClientConsumedCookies] [-requirePersistentCookie]

---

show vpn clientlessAccessProfile

Displays information about all the configured clientless access profiles, or displays detailed information about the specified clientless access profile.

Synopsys

show vpn clientlessAccessProfile [<profileName>]

Arguments

profileName

Name of the clientless access profile for which to display detailed information.

summary

fullValues

format

level

Outputs

stateflag

URLRewritePolicyLabel

Name of the configured URL rewrite policy label. If you do not specify a policy label name, then URLs are not rewritten.

JavaScriptRewritePolicyLabel

Name of the configured JavaScript rewrite policy label. If you do not specify a policy label name, then JAVA scripts are not rewritten.

CSSRewritePolicyLabel

The configured CSS rewrite policylabel.

XMLRewritePolicyLabel

The configured XML rewrite policylabel.

XComponentRewritePolicyLabel

The configured X-Component rewrite policylabel.

ReqHdrRewritePolicyLabel

Name of the configured Request rewrite policy label. If you do not specify a policy label name, then requests are not rewritten.

ResHdrRewritePolicyLabel

Name of the configured Response rewrite policy label.

RegexForFindingURLinJavaScript

Name of the pattern set that contains the regular expressions, which match the URL in Java script.

RegexForFindingURLinCSS

Name of the pattern set that contains the regular expressions, which match the URL in the CSS.

RegexForFindingURLinXComponent

Name of the pattern set that contains the regular expressions, which match the URL in X Component.

RegexForFindingURLinXML

Name of the pattern set that contains the regular expressions, which match the URL in XML.

RegexForFindingCustomURLs

Name of the pattern set that contains the regular expressions, which match the URLs in the custom content type other than HTML, CSS, XML, XCOMP, and JavaScript. The custom content type should be included in the patset ns_cvpn_custom_content_types.

ClientConsumedCookies

Specify the name of the pattern set containing the names of the cookies, which are allowed between the client and the server. If a pattern set is not specified, Access Gateway does not allow any cookies between the client and the server. A cookie that is not specified in the pattern set is handled by Access Gateway on behalf of the client.

requirePersistentCookie

Specify whether a persistent session cookie is set and accepted for clientless access. If this parameter is set to ON, COM objects, such as MSOffice, which are invoked by the browser can access the files using clientless access. Use caution because the persistent cookie is stored on the disk.

isDefault

A value of true is returned if it is a default vpnclientlessrwprofile.

description

Description of the clientless access profile.

devno

count

vpn formSSOAction

Sep 22, 2015

The following operations can be performed on "vpn formSSOAction":

add | rm | set | unset | show

add vpn formSSOAction

Creates a form-based single sign-on profile. Form based single sign-on allows users to log on one time to all protected applications in your network. Users can access web applications that require an HTML form-based logon without having to type their password again.

Synopsys

add vpn formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule <expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)]

Arguments

name

Name for the form based single sign-on profile.

actionURL

Root-relative URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Use a frequently used expression or create a custom expression describing the action that the form-based single sign-on profile takes when invoked by a policy. Used for verifying successful single sign-on.

nameValuePair

Other name-value pair attributes to send to the server, in addition to sending the user name and password. Value names are separated by an ampersand (&), such as in name1=value1&name2=value2.

responsesize

Maximum number of bytes to allow in the response size. Specifies the number of bytes in the response to be parsed for extracting the forms.

Default value: 8096

nvtype

How to process the name-value pair. Available settings function as follows:

* STATIC - The administrator-configured values are used.

* DYNAMIC - The response is parsed, the form is extracted, and then submitted.

Possible values: STATIC, DYNAMIC

Default value: NS_ACT_FSSO_NV_DYNAMIC

submitMethod

HTTP method (GET or POST) used by the single sign-on form to send the logon credentials to the logon server.

Possible values: GET, POST

Default value: NS_ACT_FSSO_SUBMIT_GET

---

rm vpn formSSOAction

Removes a configured form-based single sign-on profile.

Synopsys

rm vpn formSSOAction <name>

Arguments

name

Name of the form-based single sign-on profile to remove.

---

set vpn formSSOAction

Modifies the parameters of an existing form-based single sign-on profile (or action).

Synopsys

set vpn formSSOAction <name> [-actionURL <URL>] [-userField <string>] [-passwdField <string>] [-ssoSuccessRule <expression>] [-responsesize <positive_integer>] [-nameValuePair <string>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)]

Arguments

name

Name for the form based single sign-on profile.

actionURL

Root-relative URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Use a frequently used expression or create a custom expression describing the action that the form-based single sign-on profile takes when invoked by a policy. Used for verifying successful single sign-on.

responsesize

Maximum number of bytes to allow in the response size. Specifies the number of bytes in the response to be parsed for extracting the forms.

Default value: 8096

nameValuePair

Other name-value pair attributes to send to the server, in addition to sending the user name and password. Value names are separated by an ampersand (&), such as in name1=value1&name2=value2.

nvtype

How to process the name-value pair. Available settings function as follows:

* STATIC - The administrator-configured values are used.

* DYNAMIC - The response is parsed, the form is extracted, and then submitted.

Possible values: STATIC, DYNAMIC

Default value: NS_ACT_FSSO_NV_DYNAMIC

submitMethod

HTTP method (GET or POST) used by the single sign-on form to send the logon credentials to the logon server.

Possible values: GET, POST

Default value: NS_ACT_FSSO_SUBMIT_GET

---

unset vpn formSSOAction

Use this command to remove vpn formSSOAction settings.Refer to the set vpn formSSOAction command for meanings of the arguments.

Synopsys

unset vpn formSSOAction <name> [-responsesize] [-nameValuePair] [-nvtype] [-submitMethod]

---

show vpn formSSOAction

Displays the attributes of a form-based single sign-on profile.

Synopsys

show vpn formSSOAction [<name>]

Arguments

name

Name of the form-based single sign-on profile.

summary

fullValues

format

level

Outputs

actionURL

Root-relative URL to which the completed form is submitted.

userField

Username field.

passwdField

Password field.

responsesize

Maximum number of bytes to allow in the response size. Specifies the number of bytes in the response to be parsed for extracting the forms.

nameValuePair

Form attributes and their values to be submitted.

nvtype

Bypass Form extraction

ssoSuccessRule

Rule to be evaluated to check whether sso succeeded or not.

submitMethod

Form Submit method.

devno

count

stateflag

vpn global

Sep 22, 2015

The following operations can be performed on "vpn global":

bind | unbind | show

bind vpn global

Binds Access Gateway entities, including policies, globally.

Synopsys

bind vpn global [-policyName <string> [-priority <positive_integer>] [-secondary] [-groupExtraction]] [-intranetDomain <string>] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL>] [-appController <URL>] [-sharefile <string>]

Arguments

policyName

Name of the policy to bind globally.

intranetDomain

Intranet domain name for single sign-on.

intranetApplication

Name of the intranet application to bind globally.

nextHopServer

Name of the next hop server to bind globally.

urlName

Name of the URL of the virtual server to bind globally.

intranetIP

Range of IP addresses in an address pool or individual IP addresses to bind globally.

staServer

Web address of the Secure Ticketing Authority (STA) server to be bound globally, in the following format:

'http(s)://FQDN/URLPATH'

appController

AppController server, in the format 'http(s)://IP/FQDN'

sharefile

Sharefile server, in the format 'IP:PORT / FQDN:PORT'

---

unbind vpn global

Unbinds Access Gateway policies to the virtual server globally.

Synopsys

unbind vpn global [-policyName <string> [-secondary] [-groupExtraction]] [-intranetDomain <string>] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL>] [-appController <URL>] [-sharefile <string>]

Arguments

policyName

Name of the policy to unbind globally.

intranetDomain

A conflicting intranet domain name to be unbound.

intranetApplication

The name of a vpn intranet application to be unbound.

nextHopServer

The name of the next hop server to be unbound globally.

urlName

The name of a vpn url to be unbound from vpn global.

intranetIP

The intranet ip address or range to be unbound.

staServer

Secure Ticketing Authority (STA) server to be removed, in the format 'http(s)://IP/FQDN/URLPATH'

appController

AppController server to be removed, in the format 'http(s)://IP/FQDN'

sharefile

Sharefile server to be removed, in the format 'IP:PORT / FQDN:PORT'

---

show vpn global

Shows the Access Gateway policies that are bound to the virtual server globally.

Synopsys

show vpn global

Arguments

summary

fullValues

format

level

Outputs

stateflag

policyName

The name of the policy.

priority

The priority of the policy.

intranetDomain

The conflicting intranet domain name.

intranetApplication

The intranet vpn application.

nextHopServer

The name of the next hop server bound to vpn global.

urlName

The intranet url.

intranetIP

The intranet ip address or range.

netmask

The intranet ip address or range's netmask.

staServer

Configured Secure Ticketing Authority (STA) server.

staAuthID

Authority ID of the STA Server. Authority ID is used to match incoming STA Tickets in the SOCKS/CGP protocol with the right STA Server.

appController

Configured AppController server.

sharefile

Configured Sharefile server, in the format IP:PORT / FQDN:PORT

type

Bindpoint to which the policy is bound

policySubType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

secondary

Bind the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only to a primary authentication server but also to a secondary authentication server. User groups are aggregated across both authentication servers. The user name must be exactly the same on both authentication servers, but the authentication servers can require different passwords.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called it primary and/or secondary authentication has succeeded.

devno

count

vpn icaConnection

Sep 22, 2015

The following operations can be performed on "vpn icaConnection":

show vpn icaConnection

Displays active connections that use the ICA proxy.

Synopsys

show vpn icaConnection [-userName <string>]

Arguments

userName

User name for which to display connections.

summary

fullValues

Outputs

domain

The domain name.

srcIP

The client IP address.

srcPort

The client port.

destIP

The CPS server IP address.

destPort

The CPS server port.

peId

Core id of the session owner

stateflag

devno

count

vpn intranetApplication

Sep 22, 2015

The following operations can be performed on "vpn intranetApplication":

add | rm | show

add vpn intranetApplication

Defines intranet applications to be made accessible through the Access Gateway.

Synopsys

add vpn intranetApplication <intranetApplication> [<protocol>] ((<destIP> [-netmask <netmask>]) | <IPRange> | <hostName> | (-clientApplication <string> ... [-spoofIIP (ON | OFF)])) [-destPort <port[-port]>] [-interception (PROXY | TRANSPARENT) [-srcIP <ip_addr>] [-srcPort <port>]]

Arguments

intranetApplication

Name of the intranet application.

protocol

Protocol used by the intranet application. If protocol is set to BOTH, TCP and UDP traffic is allowed.

Possible values: TCP, UDP, ANY

destIP

Destination IP address, IP range, or host name of the intranet application. This address is the server IP address.

clientApplication

Names of the client applications, such as PuTTY and Xshell.

destPort

Destination TCP or UDP port number for the intranet application. Use a hyphen to specify a range of port numbers, for example 90-95.

Minimum value: 1

interception

Interception mode for the intranet application or resource. Correct value depends on the type of client software used to make connections. If the interception mode is set to TRANSPARENT, users connect with the Access Gateway Plug-in for Windows. With the PROXY setting, users connect with the Access Gateway Plug-in for Java.

Possible values: PROXY, TRANSPARENT

srcIP

Source IP address. Required if interception mode is set to PROXY. Default is the loopback address, 127.0.0.1.

srcPort

Source port for the application for which the Access Gateway virtual server proxies the traffic. If users are connecting from a device that uses the Access Gateway Plug-in for Java, applications must be configured manually by using the source IP address and TCP port values specified in the intranet application profile. If a port value is not set, the destination port value is used.

Minimum value: 1

---

rm vpn intranetApplication

Removes a configured intranet resource.

Synopsys

rm vpn intranetApplication <intranetApplication>

Arguments

intranetApplication

Name of the intranet resource to remove.

---

show vpn intranetApplication

Displays information about all the configured intranet resources, or displays detailed information about the specified intranet resource.

Synopsys

show vpn intranetApplication [<intranetApplication>]

Arguments

intranetApplication

Name of the intranet resource for which to display detailed information.

summary

fullValues

format

level

Outputs

protocol

The IP protocol, e.g. TCP, UDP or ANY

destIP

The destination IP address.

netmask

The destination netmask.

IPAddress

The IP address for the application. This address is the real application server IP address.

hostName

Name based interception. Names should be valid dns or wins names and will be resolved during interception on the sslvpn.

destPort

The destination port.

clientApplication

Names of the client applications, such as PuTTY and Xshell.

spoofIIP

This specifies whether to spoof this application on the client.

interception

The interception type, e.g. proxy or transparent.

srcIP

The source IP address.

srcPort

The source port.

stateflag

devno

count

vpn nextHopServer

Sep 22, 2015

The following operations can be performed on "vpn nextHopServer":

add | rm | show

add vpn nextHopServer

Enables an Access Gateway appliance in the first DMZ to communicate with one or more Access Gateway appliances in the second DMZ.

Synopsys

add vpn nextHopServer <name> <nextHopIP> <nextHopPort> [-secure (ON | OFF)]

Arguments

name

Name for the Access Gateway appliance in the first DMZ.

Maximum value: 32

nextHopIP

IP address or FQDN of the Access Gateway proxy in the second DMZ.

nextHopPort

Port number of the Access Gateway proxy in the second DMZ.

Minimum value: 1

Maximum value: 65535

secure

Use of a secure port, such as 443, for the double-hop configuration.

Possible values: ON, OFF

Default value: OFF

Example

```
add vpn nexthopserver dh1 10.1.1.1 80 -secure OFF
```

---

rm vpn nextHopServer

Removes a configured next hop server.

Synopsys

rm vpn nextHopServer <name>

Arguments

name

Name of the next hop server to remove.

Maximum value: 32

Example

rm vpn nexthopserver dh1

---

show vpn nextHopServer

Displays information about all the configured next Access Gateway hop servers, or detailed information about the specified Access Gateway next hop server.

Synopsys

show vpn nextHopServer [<name>]

Arguments

name

Name of the Access Gateway next hop server for which to display detailed information.

Maximum value: 32

summary

fullValues

format

level

Outputs

nextHopIP

Next hop IP address.

nextHopPort

Next hop port number.

secure

Next hop over secure connection.

stateflag

devno

count

Example

show vpn nexthopserver dh1

vpn parameter

Sep 22, 2015

The following operations can be performed on "vpn parameter":

set | unset | show

set vpn parameter

Sets global parameters for Access Gateway.

Synopsys

set vpn parameter [-httpPort <port> ...] [-winsIP <ip_addr>] [-dnsVserverName <string>] [-splitDns <splitDns>] [-sessTimeout <mins>] [-clientSecurity <expression> [-clientSecurityGroup <string>] [-clientSecurityMessage <string>]] [-clientSecurityLog (ON | OFF)] [-splitTunnel <splitTunnel>] [-localLanAccess (ON | OFF)] [-rfc1918 (ON | OFF)] [-spoofIIP (ON | OFF)] [-killConnections (ON | OFF)] [-transparentInterception (ON | OFF)] [-defaultAuthorizationAction (ALLOW | DENY)] [-authorizationGroup <string>] [-clientIdleTimeout <mins>] [-proxy <proxy>] [-allProtocolProxy <string> | -httpProxy <string> | -ftpProxy <string> | -socksProxy <string> | -gopherProxy <string> | -sslProxy <string>] [-proxyException <string>] [-proxyLocalBypass (ENABLED | DISABLED)] [-clientCleanupPrompt (ON | OFF)] [-forceCleanup <forceCleanup> ...] [-clientOptions <clientOptions> ...] [-clientConfiguration <clientConfiguration> ...] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-windowsAutoLogon (ON | OFF)] [-useMIP (NS | OFF)] [-useIIP <useIIP>] [-clientDebug <clientDebug>] [-loginScript <input_filename>] [-logoutScript <input_filename>] [-homePage <URL>] [-icaProxy (ON | OFF)] [-wihome <URL>] [-citrixReceiverHome <URL>] [-wiPortalMode (NORMAL | COMPACT)] [-ClientChoices (ON | OFF)] [-iipDnsSuffix <string>] [-forcedTimeout <mins>] [-forcedTimeoutWarning <mins>] [-ntDomain <string>] [-clientlessVpnMode <clientlessVpnMode>] [-clientlessModeUrlEncoding <clientlessModeUrlEncoding>] [-clientlessPersistentCookie <clientlessPersistentCookie>] [-emailHome <URL>] [-allowedLoginGroups <string>] [-encryptCsecExp (ENABLED | DISABLED)] [-appTokenTimeout <positive_integer>] [-mdxTokenTimeout <positive_integer>] [-UITHEME <UITHEME>] [-SecureBrowse (ENABLED | DISABLED)] [-storefronturl <string>] [-kcdAccount <string>]

Arguments

httpPort

Destination port numbers other than port 80, added as a comma-separated list. Traffic to these ports is processed as HTTP traffic, which allows functionality, such as HTTP authorization and single sign-on to a web application to work.

Minimum value: 1

winsIP

WINS server IP address to add to Access Gateway for name resolution.

dnsVserverName

Name of the DNS virtual server for the user session.

splitDns

Route the DNS requests to the local DNS server configured on the user device, or Access Gateway (remote), or both.

Possible values: LOCAL, REMOTE, BOTH

sessTimeout

Number of minutes after which the session times out.

Default value: 30

Minimum value: 1

clientSecurity

Specify the client security check for the user device to permit an Access Gateway session. The web address or IP address is not included in the expression for the client security check.

clientSecurityLog

Set the logging of client security checks.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_ON

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in Access Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through Access Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through Access Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the Access Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

Possible values: ON, OFF, REVERSE

Default value: VPN_SESS_ACT_OFF

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN

tunnel when the local LAN access feature is enabled:

* 10.*.*.*,

* 172.16.*.*,

* 192.168.*.*

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

spoofIIP

Indicate whether or not the application requires IP spoofing, which routes the connection to the intranet application through the virtual adapter.

Possible values: ON, OFF

Default value: ON

killConnections

Specify whether the Access Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to Access Gateway, and prevent new incoming connections on the Access Gateway Plug-in for Windows and MAC when the user is connected to Access Gateway and split tunneling is disabled.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the Access Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the Access Gateway Plug-in for Java, set this parameter to OFF.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_ON

windowsClientType

The Windows client type. Choose between two types of Windows Client\\

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed\\

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.

Possible values: AGENT, PLUGIN

Default value: VPN_SESS_ACT_CLT_AGENT

defaultAuthorizationAction

Specify the network resources that users have access to when they log on to the internal network. The default setting for authorization is to deny access to all network resources. Citrix recommends using the default global setting and then creating authorization policies to define the network resources users can access. If you set the default authorization policy to DENY, you must explicitly authorize access to any network resource, which improves security.

Possible values: ALLOW, DENY

authorizationGroup

Comma-separated list of groups in which the user is placed when none of the groups that the user is a part of is configured on Access Gateway. The authorization policy can be bound to these groups to control access to the resources.

clientIdleTimeout

Time, in minutes, after which to time out the user session if Access Gateway does not detect mouse or keyboard activity.

Minimum value: 1

Maximum value: 9999

proxy

Set options to apply proxy for accessing the internal resources. Available settings function as follows:

* BROWSER - Proxy settings are configured only in Internet Explorer and Firefox browsers.

* NS - Proxy settings are configured on the NetScaler appliance.

* OFF - Proxy settings are not configured.

Possible values: BROWSER, NS, OFF

allProtocolProxy

IP address of the proxy server to use for all protocols supported by Access Gateway.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

Possible values: ENABLED, DISABLED

Default value: VPN_SESS_ACT_DISABLED

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_ON

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

Display only the configured menu options when you select the "Configure Access Gateway" option in the Access Gateway Plug-in's system tray icon for Windows.

clientConfiguration

Display only the configured tabs when you select the "Configure Access Gateway" option in the Access Gateway Plug-in's system tray icon for Windows.

SSO

Set single sign-on (SSO) for the session. When the user accesses a server, the user's logon credentials are passed to the server for authentication.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

Possible values: PRIMARY, SECONDARY

Default value: VPN_SESS_ACT_USE_PRIMARY_CREDENTIALS

windowsAutoLogon

Enable or disable the Windows Auto Logon for the session. If a VPN session is established after this setting is enabled, the user is automatically logged on by using Windows credentials after the system is restarted.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

useMIP

Enable or disable the use of a unique IP address alias, or a mapped IP address, as the client IP address for each client session. Allow Access Gateway to use the mapped IP address as an intranet IP address when all other IP addresses are not available.

When IP pooling is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

Possible values: NS, OFF

Default value: VPN_SESS_ACT_NS

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILLOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

Possible values: NOSPILLOVER, SPILLOVER, OFF

Default value: VPN_SESS_ACT_NOSPILLOVER

clientDebug

Set the trace level on Access Gateway. Technical support technicians use these debug logs for in-depth debugging and troubleshooting purposes. Available settings function as follows:

* DEBUG - Detailed debug messages are collected and written into the specified file.

* STATS - Application audit level error messages and debug statistic counters are written into the specified file.

* EVENTS - Application audit-level error messages are written into the specified file.

* OFF - Only critical events are logged into the Windows Application Log.

Possible values: debug, stats, events, OFF

Default value: VPN_FLAG_TRACE_OFF

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

homePage

Web address of the home page that appears when users log on. Otherwise, users receive the default home page for Access Gateway, which is the Access Interface.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the Access Gateway Plug-in.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

wihome

Web address of the Web Interface server, such as http://<ipAddress>/Citrix/XenApp, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure Access Gateway so that when users log on to the appliance, the Access Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

Possible values: NORMAL, COMPACT

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the Access Gateway Plug-in for Windows, Access Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how Access Gateway is configured, users are presented with up to three icons for logon choices. The most common are the Access Gateway Plug-in for Windows, Web Interface, and clientless access.

Possible values: ON, OFF

Default value: VPN_SESS_ACT_OFF

epaClientType

Choose between two types of End point Windows Client

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.

Possible values: AGENT, PLUGIN

iipDnsSuffix

An intranet IP DNS suffix. When a user logs on to Access Gateway and is assigned an IP address, a DNS record for the user name and IP address combination is added to Access Gateway DNS cache. You can configure a DNS suffix to append to the user name when the DNS record is added to the cache. You can reach to the host from where the user is logged on by using the user's name, which can be easier to remember than an IP address. When the user logs off from Access Gateway, the record is removed from the DNS cache.

forcedTimeout

Force a disconnection from the Access Gateway Plug-in with Access Gateway after a specified number of minutes. If the session closes, the user must log on again.

Minimum value: 1

Maximum value: 65535

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

Minimum value: 1

Maximum value: 255

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Enable clientless access for web, XenApp or XenDesktop, and FileShare resources without installing the Access Gateway Plug-in. Available settings function as follows:

* ON - Allow only clientless access.

* OFF - Allow clientless access after users log on with the Access Gateway Plug-in.

* DISABLED - Do not allow clientless access.

Possible values: ON, OFF, DISABLED

Default value: VPN_SESS_ACT_CVPNMODE_OFF

clientlessModeUrlEncoding

When clientless access is enabled, you can choose to encode the addresses of internal web applications or to leave the address as clear text. Available settings function as follows:

* OPAQUE - Use standard encoding mechanisms to make the domain and protocol part of the resource unclear to users.

* CLEAR - Do not encode the web address and make it visible to users.

* ENCRYPT - Allow the domain and protocol to be encrypted using a session key. When the web address is encrypted, the URL is different for each user session for the same web resource. If users bookmark the encoded web address, save it in the web browser and then log off, they cannot connect to the web address when they log on and use the bookmark. If users save the encrypted bookmark in the Access Interface during their session, the bookmark works each time the user logs on.

Possible values: TRANSPARENT, OPAQUE, ENCRYPT

Default value: VPN_SESS_ACT_CVPN_ENC_OPAQUE

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. Access Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

Possible values: ALLOW, DENY, PROMPT

Default value: VPN_SESS_ACT_CVPN_PERSCOOKIE_DENY

emailHome

Web address for the web-based email, such as Outlook Web Access.

allowedLoginGroups

Specify groups that have permission to log on to Access Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

encryptCsecExp

Enable encryption of client security expressions.

Possible values: ENABLED, DISABLED

Default value: VPN_SESS_ACT_ENABLED

appTokenTimeout

The timeout value in seconds for tokens to access cloud gateway applications

Default value: 100

Minimum value: 1

Maximum value: 255

mdxTokenTimeout

Validity of MDX Token in minutes. This token is used for mdx services to access backend and valid HEAD and GET request.

Default value: 10

Minimum value: 1

Maximum value: 1440

UITHEME

Set VPN UI Theme to Green-Bubble, Caxton or Custom; default is Caxton.

Possible values: DEFAULT, GREENBUBBLE, CUSTOM

SecureBrowse

Allow users to connect through Access Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

Possible values: ENABLED, DISABLED

Default value: VPN_SESS_ACT_ENABLED

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

Example

set vpn parameter -httpport 80 90 -winsIP 192.168.0.220 -dnsVserverName mydns -sessTimeout 240

---

unset vpn parameter

Removes global parameters for Access Gateway..Refer to the set vpn parameter command for meanings of the arguments.

Synopsys

unset vpn parameter [-httpPort] [-winsIP] [-dnsVserverName] [-splitDns] [-sessTimeout] [-clientSecurity] [-clientSecurityGroup] [-clientSecurityMessage] [-clientSecurityLog] [-authorizationGroup] [-clientIdleTimeout] [-allProtocolProxy | -httpProxy | -ftpProxy | -socksProxy | -gopherProxy | -sslProxy] [-proxyException] [-forceCleanup] [-clientOptions] [-clientConfiguration] [-loginScript] [-logoutScript] [-homePage] [-proxy] [-wihome] [-citrixReceiverHome] [-wiPortalMode] [-iipDnsSuffix] [-forcedTimeout] [-forcedTimeoutWarning] [-defaultAuthorizationAction] [-ntDomain] [-clientlessVpnMode] [-emailHome] [-clientlessModeUrlEncoding] [-clientlessPersistentCookie] [-allowedLoginGroups] [-appTokenTimeout] [-mdxTokenTimeout] [-storefronturl] [-UITHEME] [-kcdAccount] [-splitTunnel] [-localLanAccess] [-rfc1918] [-spoofIIP] [-killConnections] [-transparentInterception] [-proxyLocalBypass] [-clientCleanupPrompt] [-SSO] [-ssoCredential] [-windowsAutoLogon] [-useMIP] [-useIIP] [-clientDebug] [-icaProxy] [-ClientChoices] [-encryptCsecExp] [-SecureBrowse]

---

show vpn parameter

Displays the configured Access Gateway parameters.

Synopsys

show vpn parameter

Arguments

format

level

Outputs

name

The VPN name.

httpPort

The HTTP Port.

winsIP

The WINS server IP address used for WINS host resolution by the VPN.

dnsVserverName

The configured DNS vserver used for DNS host resolution by the VPN.

splitDns

The VPN client SplitDns state.

sessTimeout

The session timeout, in minutes.

clientSecurity

The client security check applied to client sessions. This is in the form of an Expression. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in Access Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through Access Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through Access Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the Access Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

rfc1918

Only allow RFC1918 local addresses when local LAN access feature is enabled.

spoofIIP

Controls the Spoofing of Intranet IP to the Windows Applications by Windows VPN client when the end-user is connected to SSL VPN in '-splittunnel OFF' mode.

killConnections

Determines whether Windows VPN client should kill all pre-existing connections (ie, the connections existing before the end user logged in to SSL VPN) and prevent new incoming connections on the Windows Client system when the end-user is connected to SSL VPN in '-splittunnel OFF' mode.

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the Access Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the Access Gateway Plug-in for Java, set this parameter to OFF.

windowsClientType

The windows client type.NOTE: This attribute is deprecated.This argument is deprecated since ActiveX is no longer supported.

defaultAuthorizationAction

The Authentication Action, e.g. allow or deny.

authorizationGroup

The authorization group applied to the session.

clientIdleTimeout

The client idle timeout, in minutes.

clientIdleTimeoutWarning

The time after which the client gets a timeout warning, in minutes.

proxy

Proxy configuration for the session.

allProtocolProxy

Address set for all proxies.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

The Proxy Exception string that is configured in the Browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

forceCleanup

Whether or not to force a cleanup on exit from the VPN session.

clientOptions

List of configured buttons(and/or menu options in the docked client) in the Windows VPN client.

clientConfiguration

List of configured tabs in the Windows VPN client.

SSO

Enable or Disable Single Sign-On.

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

windowsAutoLogon

Enable or Disable Windows Auto Logon.

useMIP

Enables or disables the use of a Mapped IP address for the session.

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILLOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

clientDebug

Whether or not to add debugging information to the activity log on the client.

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

homePage

The home page URL, or 'none'. 'none' is case sensitive.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the Access Gateway Plug-in.

wihome

Web address of the Web Interface server, such as http://<ipAddress>/Citrix/XenApp, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure Access Gateway so that when users log on to the appliance, the Access Gateway Plug-in opens a web browser that allows single sign-on to the Citrix

Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the Access Gateway Plug-in for Windows, Access Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how Access Gateway is configured, users are presented with up to three icons for logon choices. The most common are the Access Gateway Plug-in for Windows, Web Interface, and clientless access.

epaClientType

Choose between two types of End point Windows Client

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.NOTE: This attribute is deprecated.This argument is not supported

iipDnsSuffix

The DNS suffix for the intranet IP address.

forcedTimeout

The time, in minutes after which a timeout is forced.

forcedTimeoutWarning

The time, in minutes, after which a timeout warning is issued.

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Whether clientlessVPN is available to the session.

clientlessModeUrlEncoding

URL encoding to be used for clientless mode.

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents

hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. Access Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

emailHome

Web address for the web-based email, such as Outlook Web Access.

allowedLoginGroups

Specify groups that have permission to log on to Access Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

encryptCsecExp

Enable encryption of client security expressions.

appTokenTimeout

The timeout value in seconds for tokens to access cloud gateway applications

mdxTokenTimeout

Validity of MDX Token in minutes. This token is used for mdx services to access backend and valid HEAD and GET request.

UITHEME

Set VPN UI Theme to Green-Bubble, Caxton or Custom; default is Caxton.

SecureBrowse

Allow users to connect through Access Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

vpn samlSSOProfile

Sep 22, 2015

The following operations can be performed on "vpn samlSSOProfile":

add | rm | set | unset | show

add vpn samlSSOProfile

Creates a SAML single sign-on profile. This profile is employed in triggering saml assertion to a target service based on traffic profile.

Synopsys

add vpn samlSSOProfile <name> -samlSigningCertName <string> -assertionConsumerServiceURL <URL> -relaystateRule <expression> [-sendPassword (ON | OFF)] [-samlIssuerName <string>]

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

relaystateRule

Expression to extract relaystate to be sent along with assertion.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

rm vpn samlSSOProfile

Deletes an existing saml single sign-on traffic profile.

Synopsys

rm vpn samlSSOProfile <name>

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

---

set vpn samlSSOProfile

Modifies the specified attributes of a saml single sign-on traffic profile.

Synopsys

set vpn samlSSOProfile <name> [-samlSigningCertName <string>] [-assertionConsumerServiceURL <URL>] [-sendPassword (ON | OFF)] [-samlIssuerName <string>] [-relaystateRule <expression>]

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion.

---

unset vpn samlSSOProfile

Use this command to remove vpn samlSSOProfile settings.Refer to the set vpn samlSSOProfile command for meanings of the arguments.

Synopsys

unset vpn samlSSOProfile <name> [-samlSigningCertName] [-sendPassword] [-samlIssuerName]

---

show vpn samlSSOProfile

Displays information about all configured saml single sign-on profiles, or displays detailed information about the specified action.

Synopsys

show vpn samlSSOProfile [<name>]

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

summary

fullValues

format

level

Outputs

samlSigningCertName

Name of the signing authority as given in the SAML server?s SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion.

devno

count

stateflag

vpn sessionAction

Sep 22, 2015

The following operations can be performed on "vpn sessionAction":

add | rm | set | unset | show

add vpn sessionAction

Adds a session profile (action) to bind to a session policy that is applied to a user session if the policy expression conditions are met.

Synopsys

add vpn sessionAction <name> [-httpPort <port> ...] [-winsIP <ip_addr>] [-dnsVserverName <string>] [-splitDns <splitDns>] [-sessTimeout <mins>] [-clientSecurity <expression> [-clientSecurityGroup <string>] [-clientSecurityMessage <string>]] [-clientSecurityLog (ON | OFF)] [-splitTunnel <splitTunnel>] [-localLanAccess (ON | OFF)] [-rfc1918 (ON | OFF)] [-spoofIIP (ON | OFF)] [-killConnections (ON | OFF)] [-transparentInterception (ON | OFF)] [-defaultAuthorizationAction (ALLOW | DENY)] [-authorizationGroup <string>] [-clientIdleTimeout <mins>] [-proxy <proxy>] [-allProtocolProxy <string> | -httpProxy <string> | -ftpProxy <string> | -socksProxy <string> | -gopherProxy <string> | -sslProxy <string>] [-proxyException <string>] [-proxyLocalBypass (ENABLED | DISABLED)] [-clientCleanupPrompt (ON | OFF)] [-forceCleanup <forceCleanup> ...] [-clientOptions <clientOptions> ...] [-clientConfiguration <clientConfiguration> ...] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-windowsAutoLogon (ON | OFF)] [-useMIP (NS | OFF)] [-useIIP <useIIP>] [-clientDebug <clientDebug>] [-loginScript <input_filename>] [-logoutScript <input_filename>] [-homePage <URL>] [-icaProxy (ON | OFF)] [-wihome <URL>] [-citrixReceiverHome <URL>] [-wiPortalMode (NORMAL | COMPACT)] [-ClientChoices (ON | OFF)] [-iipDnsSuffix <string>] [-forcedTimeout <mins>] [-forcedTimeoutWarning <mins>] [-ntDomain <string>] [-clientlessVpnMode <clientlessVpnMode>] [-emailHome <URL>] [-clientlessModeUrlEncoding <clientlessModeUrlEncoding>] [-clientlessPersistentCookie <clientlessPersistentCookie>] [-allowedLoginGroups <string>] [-SecureBrowse (ENABLED | DISABLED)] [-storefronturl <string>] [-kcdAccount <string>]

Arguments

name

Name for the Access Gateway profile (action). Must begin with an ASCII alphabetic or underscore (_) character, and must consist only of ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

httpPort

Destination port numbers other than port 80, added as a comma-separated list. Traffic to these ports is processed as HTTP traffic, which allows functionality, such as HTTP authorization and single sign-on to a web application to work.

Minimum value: 1

winsIP

WINS server IP address to add to Access Gateway for name resolution.

dnsVserverName

Name of the DNS virtual server for the user session.

splitDns

Route the DNS requests to the local DNS server configured on the user device, or Access Gateway (remote), or both.

Possible values: LOCAL, REMOTE, BOTH

sessTimeout

Number of minutes after which the session times out.

Minimum value: 1

clientSecurity

Specify the client security check for the user device to permit an Access Gateway session. The web address or IP address is not included in the expression for the client security check.

clientSecurityLog

Set the logging of client security checks.

Possible values: ON, OFF

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in Access Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through Access Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through Access Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the Access Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

Possible values: ON, OFF, REVERSE

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

Possible values: ON, OFF

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.*.*,

* 172.16.*.*,

* 192.168.*.*

Possible values: ON, OFF

spoofIIP

IP address that the intranet application uses to route the connection through the virtual adapter.

Possible values: ON, OFF

killConnections

Specify whether the Access Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to Access Gateway, and prevent new incoming connections on the Access Gateway Plug-in for Windows and MAC when the user is connected to Access Gateway and split tunneling is disabled.

Possible values: ON, OFF

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the Access Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the Access Gateway Plug-in for Java, set this parameter to OFF.

Possible values: ON, OFF

windowsClientType

Choose between two types of Windows Client\\

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed\\

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.

Possible values: AGENT, PLUGIN

defaultAuthorizationAction

Specify the network resources that users have access to when they log on to the internal network. The default setting for authorization is to deny access to all network resources. Citrix recommends using the default global setting and then creating authorization policies to define the network resources users can access. If you set the default authorization policy to DENY, you must explicitly authorize access to any network resource, which improves

security.

Possible values: ALLOW, DENY

authorizationGroup

Comma-separated list of groups in which the user is placed when none of the groups that the user is a part of is configured on Access Gateway. The authorization policy can be bound to these groups to control access to the resources.

clientIdleTimeout

Time, in minutes, after which to time out the user session if Access Gateway does not detect mouse or keyboard activity.

Minimum value: 1

Maximum value: 9999

proxy

Set options to apply proxy for accessing the internal resources. Available settings function as follows:

* BROWSER - Proxy settings are configured only in Internet Explorer and Firefox browsers.

* NS - Proxy settings are configured on the NetScaler appliance.

* OFF - Proxy settings are not configured.

Possible values: BROWSER, NS, OFF

allProtocolProxy

IP address of the proxy server to use for all protocols supported by Access Gateway.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

Possible values: ENABLED, DISABLED

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

Possible values: ON, OFF

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

Display only the configured menu options when you select the "Configure Access Gateway" option in the Access Gateway Plug-in's system tray icon for Windows.

clientConfiguration

Display only the configured tabs when you select the "Configure Access Gateway" option in the Access Gateway Plug-in's system tray icon for Windows.

SSO

Set single sign-on (SSO) for the session. When the user accesses a server, the user's logon credentials are passed to the server for authentication.

Possible values: ON, OFF

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

Possible values: PRIMARY, SECONDARY

windowsAutoLogon

Enable or disable the Windows Auto Logon for the session. If a VPN session is established after this setting is enabled, the user is automatically logged on by using Windows credentials after the system is restarted.

Possible values: ON, OFF

useMIP

Enable or disable the use of a unique IP address alias, or a mapped IP address, as the client IP address for each client session. Allow Access Gateway to use the mapped IP address as an intranet IP address when all other IP addresses are not available.

When IP pooling is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

Possible values: NS, OFF

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILLOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

Possible values: NOSPILLOVER, SPILLOVER, OFF

clientDebug

Set the trace level on Access Gateway. Technical support technicians use these debug logs for in-depth debugging and troubleshooting purposes. Available settings function as follows:

* DEBUG - Detailed debug messages are collected and written into the specified file.

* STATS - Application audit level error messages and debug statistic counters are written into the specified file.

* EVENTS - Application audit-level error messages are written into the specified file.

* OFF - Only critical events are logged into the Windows Application Log.

Possible values: debug, stats, events, OFF

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

homePage

Web address of the home page that appears when users log on. Otherwise, users receive the default home page for Access Gateway, which is the Access Interface.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the Access Gateway Plug-in.

Possible values: ON, OFF

wihome

Web address of the Web Interface server, such as http://<ipAddress>/Citrix/XenApp, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure Access Gateway so that when users log on to the appliance, the Access Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

Possible values: NORMAL, COMPACT

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the Access Gateway Plug-in for Windows, Access Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how Access Gateway is configured, users are presented with up to three icons for logon choices. The most common are the Access Gateway Plug-in for Windows, Web Interface, and clientless access.

Possible values: ON, OFF

epaClientType

Choose between two types of End point Windows Client

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.

Possible values: AGENT, PLUGIN

iipDnsSuffix

An intranet IP DNS suffix. When a user logs on to Access Gateway and is assigned an IP address, a DNS record for the user name and IP address combination is added to Access Gateway DNS cache. You can configure a DNS suffix

to append to the user name when the DNS record is added to the cache. You can reach to the host from where the user is logged on by using the user's name, which can be easier to remember than an IP address. When the user logs off from Access Gateway, the record is removed from the DNS cache.

forcedTimeout

Force a disconnection from the Access Gateway Plug-in with Access Gateway after a specified number of minutes. If the session closes, the user must log on again.

Minimum value: 1

Maximum value: 65535

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

Minimum value: 1

Maximum value: 255

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Enable clientless access for web, XenApp or XenDesktop, and FileShare resources without installing the Access Gateway Plug-in. Available settings function as follows:

* ON - Allow only clientless access.

* OFF - Allow clientless access after users log on with the Access Gateway Plug-in.

* DISABLED - Do not allow clientless access.

Possible values: ON, OFF, DISABLED

emailHome

Web address for the web-based email, such as Outlook Web Access.

clientlessModeUrlEncoding

When clientless access is enabled, you can choose to encode the addresses of internal web applications or to leave the address as clear text. Available settings function as follows:

* OPAQUE - Use standard encoding mechanisms to make the domain and protocol part of the resource unclear to users.

* CLEAR - Do not encode the web address and make it visible to users.

* ENCRYPT - Allow the domain and protocol to be encrypted using a session key. When the web address is

encrypted, the URL is different for each user session for the same web resource. If users bookmark the encoded web address, save it in the web browser and then log off, they cannot connect to the web address when they log on and use the bookmark. If users save the encrypted bookmark in the Access Interface during their session, the bookmark works each time the user logs on.

Possible values: TRANSPARENT, OPAQUE, ENCRYPT

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. Access Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

Possible values: ALLOW, DENY, PROMPT

allowedLoginGroups

Specify groups that have permission to log on to Access Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

SecureBrowse

Allow users to connect through Access Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

Possible values: ENABLED, DISABLED

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

---

rm vpn sessionAction

Removes an action that was previously added to a session policy.

Synopsys

rm vpn sessionAction <name>

Arguments

name

Name of the action to remove.

---

set vpn sessionAction

Modifies an action that was previously added to a session policy that is applied to a user session if the policy expression conditions are met.

Synopsys

set vpn sessionAction <name> [-httpPort <port> ...] [-winsIP <ip_addr>] [-dnsVserverName <string>] [-splitDns <splitDns>] [-sessTimeout <mins>] [-clientSecurity <expression> [-clientSecurityGroup <string>] [-clientSecurityMessage <string>]] [-clientSecurityLog (ON | OFF)] [-splitTunnel <splitTunnel>] [-localLanAccess (ON | OFF)] [-rfc1918 (ON | OFF)] [-spoofIIP (ON | OFF)] [-killConnections (ON | OFF)] [-transparentInterception (ON | OFF)] [-defaultAuthorizationAction (ALLOW | DENY)] [-authorizationGroup <string>] [-clientIdleTimeout <mins>] [-proxy <proxy>] [-allProtocolProxy <string> | -httpProxy <string> | -ftpProxy <string> | -socksProxy <string> | -gopherProxy <string> | -sslProxy <string>] [-proxyException <string>] [-proxyLocalBypass (ENABLED | DISABLED)] [-clientCleanupPrompt (ON | OFF)] [-forceCleanup <forceCleanup> ...] [-clientOptions <clientOptions> ...] [-clientConfiguration <clientConfiguration> ...] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-windowsAutoLogon (ON | OFF)] [-useMIP (NS | OFF)] [-useIIP <useIIP>] [-clientDebug <clientDebug>] [-loginScript <input_filename>] [-logoutScript <input_filename>] [-homePage <URL>] [-icaProxy (ON | OFF)] [-wihome <URL>] [-citrixReceiverHome <URL>] [-wiPortalMode (NORMAL | COMPACT)] [-ClientChoices (ON | OFF)] [-iipDnsSuffix <string>] [-forcedTimeout <mins>] [-forcedTimeoutWarning <mins>] [-ntDomain <string>] [-clientlessVpnMode <clientlessVpnMode>] [-emailHome <URL>] [-clientlessModeUrlEncoding <clientlessModeUrlEncoding>] [-clientlessPersistentCookie <clientlessPersistentCookie>] [-allowedLoginGroups <string>] [-SecureBrowse (ENABLED | DISABLED)] [-storefronturl <string>] [-kcdAccount <string>]

Arguments

name

The name of the vpn session action.

httpPort

Destination port numbers other than port 80, added as a comma-separated list. Traffic to these ports is processed as HTTP traffic, which allows functionality, such as HTTP authorization and single sign-on to a web application to work.

Minimum value: 1

winsIP

The WINS server ip address.

dnsVserverName

Name of the DNS virtual server for the user session.

splitDns

Route the DNS requests to the local DNS server configured on the user device, or Access Gateway (remote), or both.

Possible values: LOCAL, REMOTE, BOTH

sessTimeout

Number of minutes after which the session times out.

Minimum value: 1

clientSecurity

Specify the client security check for the user device to permit an Access Gateway session. The web address or IP address is not included in the expression for the client security check.

clientSecurityLog

Set the logging of client security checks.

Possible values: ON, OFF

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in Access Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through Access Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through Access Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the Access Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

Possible values: ON, OFF, REVERSE

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

Possible values: ON, OFF

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.*.*,

* 172.16.*.*,

* 192.168.*.*

Possible values: ON, OFF

spoofIIP

IP address that the intranet application uses to route the connection through the virtual adapter.

Possible values: ON, OFF

killConnections

Specify whether the Access Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to Access Gateway, and prevent new incoming connections on the Access Gateway Plug-in for Windows and MAC when the user is connected to Access Gateway and split tunneling is disabled.

Possible values: ON, OFF

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the Access Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the Access Gateway Plug-in for Java, set this parameter to OFF.

Possible values: ON, OFF

windowsClientType

Choose between two types of Windows Client\\

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed\\

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.

Possible values: AGENT, PLUGIN

defaultAuthorizationAction

Specify the network resources that users have access to when they log on to the internal network. The default setting for authorization is to deny access to all network resources. Citrix recommends using the default global setting and then creating authorization policies to define the network resources users can access. If you set the default authorization policy to DENY, you must explicitly authorize access to any network resource, which improves security.

Possible values: ALLOW, DENY

authorizationGroup

Comma-separated list of groups in which the user is placed when none of the groups that the user is a part of is

configured on Access Gateway. The authorization policy can be bound to these groups to control access to the resources.

clientIdleTimeout

Time, in minutes, after which to time out the user session if Access Gateway does not detect mouse or keyboard activity.

Minimum value: 1

Maximum value: 9999

proxy

Set options to apply proxy for accessing the internal resources. Available settings function as follows:

* BROWSER - Proxy settings are configured only in Internet Explorer and Firefox browsers.

* NS - Proxy settings are configured on the NetScaler appliance.

* OFF - Proxy settings are not configured.

Possible values: BROWSER, NS, OFF

allProtocolProxy

IP address of the proxy server to use for all protocols supported by Access Gateway.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

Possible values: ENABLED, DISABLED

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

Possible values: ON, OFF

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

Display only the configured menu options when you select the "Configure Access Gateway" option in the Access Gateway Plug-in's system tray icon for Windows.

clientConfiguration

Display only the configured tabs when you select the "Configure Access Gateway" option in the Access Gateway Plug-in's system tray icon for Windows.

SSO

Set single sign-on (SSO) for the session. When the user accesses a server, the user's logon credentials are passed to the server for authentication.

Possible values: ON, OFF

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

Possible values: PRIMARY, SECONDARY

windowsAutoLogon

Enable or disable the Windows Auto Logon for the session. If a VPN session is established after this setting is enabled, the user is automatically logged on by using Windows credentials after the system is restarted.

Possible values: ON, OFF

useMIP

Enable or disable the use of a unique IP address alias, or a mapped IP address, as the client IP address for each client session. Allow Access Gateway to use the mapped IP address as an intranet IP address when all other IP addresses are not available.

When IP pooling is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

Possible values: NS, OFF

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILLOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

Possible values: NOSPILLOVER, SPILLOVER, OFF

clientDebug

Set the trace level on Access Gateway. Technical support technicians use these debug logs for in-depth debugging and troubleshooting purposes. Available settings function as follows:

* DEBUG - Detailed debug messages are collected and written into the specified file.

* STATS - Application audit level error messages and debug statistic counters are written into the specified file.

* EVENTS - Application audit-level error messages are written into the specified file.

* OFF - Only critical events are logged into the Windows Application Log.

Possible values: debug, stats, events, OFF

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

homePage

Web address of the home page that appears when users log on. Otherwise, users receive the default home page for Access Gateway, which is the Access Interface.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the Access Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

wihome

Web address of the Web Interface server, such as http://<ipAddress>/Citrix/XenApp, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure Access Gateway so that when users log on to the appliance, the Access Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

Possible values: NORMAL, COMPACT

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the Access Gateway Plug-in for Windows, Access Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how Access Gateway is configured, users are presented with up to three icons for logon choices. The most common are the Access Gateway Plug-in for Windows, Web Interface, and clientless access.

Possible values: ON, OFF

epaClientType

Choose between two types of End point Windows Client

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.

Possible values: AGENT, PLUGIN

iipDnsSuffix

An intranet IP DNS suffix. When a user logs on to Access Gateway and is assigned an IP address, a DNS record for the user name and IP address combination is added to Access Gateway DNS cache. You can configure a DNS suffix to append to the user name when the DNS record is added to the cache. You can reach to the host from where the user is logged on by using the user's name, which can be easier to remember than an IP address. When the user logs off from Access Gateway, the record is removed from the DNS cache.

forcedTimeout

Force a disconnection from the Access Gateway Plug-in with Access Gateway after a specified number of minutes. If the session closes, the user must log on again.

Minimum value: 1

Maximum value: 65535

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

Minimum value: 1

Maximum value: 255

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Enable clientless access for web, XenApp or XenDesktop, and FileShare resources without installing the Access Gateway Plug-in. Available settings function as follows:

* ON - Allow only clientless access.

* OFF - Allow clientless access after users log on with the Access Gateway Plug-in.

* DISABLED - Do not allow clientless access.

Possible values: ON, OFF, DISABLED

Default value: VPN_SESS_ACT_CVPNMODE_OFF

emailHome

Web address for the web-based email, such as Outlook Web Access.

clientlessModeUrlEncoding

When clientless access is enabled, you can choose to encode the addresses of internal web applications or to leave the address as clear text. Available settings function as follows:

* OPAQUE - Use standard encoding mechanisms to make the domain and protocol part of the resource unclear to users.

* CLEAR - Do not encode the web address and make it visible to users.

* ENCRYPT - Allow the domain and protocol to be encrypted using a session key. When the web address is encrypted, the URL is different for each user session for the same web resource. If users bookmark the encoded web address, save it in the web browser and then log off, they cannot connect to the web address when they log on and use the bookmark. If users save the encrypted bookmark in the Access Interface during their session, the

bookmark works each time the user logs on.

Possible values: TRANSPARENT, OPAQUE, ENCRYPT

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. Access Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

Possible values: ALLOW, DENY, PROMPT

Default value: VPN_SESS_ACT_CVPN_PERSCOOKIE_DENY

allowedLoginGroups

Specify groups that have permission to log on to Access Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

SecureBrowse

Allow users to connect through Access Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

Possible values: ENABLED, DISABLED

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

---

unset vpn sessionAction

Use this command to remove vpn sessionAction settings.Refer to the set vpn sessionAction command for meanings of the arguments.

Synopsys

unset vpn sessionAction <name> [-httpPort] [-winsIP] [-dnsVserverName] [-splitDns] [-sessTimeout] [-clientSecurity] [-

clientSecurityGroup] [-clientSecurityMessage] [-clientSecurityLog] [-splitTunnel] [-localLanAccess] [-rfc1918] [-spoofIIP] [-killConnections] [-transparentInterception] [-defaultAuthorizationAction] [-authorizationGroup] [-clientIdleTimeout] [-proxy] [-allProtocolProxy] [-httpProxy] [-ftpProxy] [-socksProxy] [-gopherProxy] [-sslProxy] [-proxyException] [-proxyLocalBypass] [-clientCleanupPrompt] [-forceCleanup] [-clientOptions] [-clientConfiguration] [-SSO] [-ssoCredential] [-windowsAutoLogon] [-useMIP] [-useIIP] [-clientDebug] [-loginScript] [-logoutScript] [-homePage] [-icaProxy] [-wihome] [-citrixReceiverHome] [-wiPortalMode] [-ClientChoices] [-iipDnsSuffix] [-forcedTimeout] [-forcedTimeoutWarning] [-ntDomain] [-clientlessVpnMode] [-emailHome] [-clientlessModeUrlEncoding] [-clientlessPersistentCookie] [-allowedLoginGroups] [-SecureBrowse] [-storefronturl] [-kcdAccount]

---

show vpn sessionAction

Displays a session action that is applied to a user session if the policy expression conditions are met.

Synopsys

show vpn sessionAction [<name>]

Arguments

name

Name of the session action to display.

summary

fullValues

format

level

Outputs

httpPort

The HTTP port for this session action

winsIP

The WINS server IP address for this session action.

dnsVserverName

The name of the DNS vserver configured by the session action.

splitDns

The VPN client SplitDns state.

sessTimeout

The session timeout, in minutes, set by the action.

clientSecurity

The client security check string being applied. This is in the form of an Expression. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in Access Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through Access Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through Access Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the Access Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.*.*,

* 172.16.*.*,

* 192.168.*.*

spoofIIP

IP address that the intranet application uses to route the connection through the virtual adapter.

killConnections

Specify whether the Access Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to Access Gateway, and prevent new incoming connections on the Access Gateway Plug-in for Windows and MAC when the user is connected to Access Gateway and split tunneling is disabled.

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the Access Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the Access Gateway Plug-in for Java, set this parameter to OFF.

windowsClientType

Windows client type, e.g. Agent or ActiveXNOTE: This attribute is deprecated.This argument is deprecated since ActiveX is no longer supported.

defaultAuthorizationAction

The Authorization Action, e.g. allow or deny

authorizationGroup

The authorization group applied to client sessions.

clientIdleTimeout

The client idle timeout, in minutes.

clientIdleTimeoutWarning

The time after which the client gets a timeout warning, in minutes.

proxy

The state of proxy configuration for the session.

allProtocolProxy

The address set for all proxies.

httpProxy

The HTTP proxy IP address.

ftpProxy

The FTP proxy IP address.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

The Gopher proxy IP address.

sslProxy

The HTTPS proxy IP address.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

List of configured buttons(and/or menu options in the docked client) in the Windows VPN client.

clientConfiguration

List of configured tabs in the Windows VPN client.

SSO

Whether or not Single Sign-On is used for this session.

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

windowsAutoLogon

Whether or not Windows Auto Logon is enabled for this session.

useMIP

Whether or not a Mapped IP address is used for the session

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILLOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

clientDebug

Trace level on the Windows VPN Client.

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "$" in the path signifies that the word following the "$" is an environment variable.

homePage

The client home page.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the Access Gateway Plug-in.

wihome

Web address of the Web Interface server, such as http://<ipAddress>/Citrix/XenApp, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does

not appear as a client choice.

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure Access Gateway so that when users log on to the appliance, the Access Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the Access Gateway Plug-in for Windows, Access Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how Access Gateway is configured, users are presented with up to three icons for logon choices. The most common are the Access Gateway Plug-in for Windows, Web Interface, and clientless access.

epaClientType

Choose between two types of End point Windows Client

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed

b) Activex Control - ActiveX control run by Microsoft's Internet Explorer.NOTE: This attribute is deprecated.This argument is not supported

iipDnsSuffix

The IntranetIP DNS suffix.

forcedTimeout

Force a disconnection from the Access Gateway Plug-in with Access Gateway after a specified number of minutes. If the session closes, the user must log on again.

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Whether clientlessVPN is available to the session.

clientlessModeUrlEncoding

URL encoding used in clientless mode.

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. Access Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

emailHome

The EMail home for the portal

stateflag

allowedLoginGroups

Specify groups that have permission to log on to Access Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

SecureBrowse

Allow users to connect through Access Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

vpn sessionPolicy

Sep 22, 2015

The following operations can be performed on "vpn sessionPolicy":

add | rm | set | unset | show

add vpn sessionPolicy

Creates a new session policy that, if bound, is applied after the user logs on to Access Gateway, and that determines the properties of the user session.

Synopsys

add vpn sessionPolicy <name> <rule> <action>

Arguments

name

Name for the new session policy that is applied after the user logs on to Access Gateway.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied by the new session policy if the rule criteria are met.

---

rm vpn sessionPolicy

Removes the session policy that is applied after the user logs on to Access Gateway.

Synopsys

rm vpn sessionPolicy <name>

Arguments

name

Name of the session policy to remove.

---

set vpn sessionPolicy

Modifies the rule or action of a session policy.

Synopsys

set vpn sessionPolicy <name> [-rule <expression>] [-action <string>]

Arguments

name

Name of the session policy to modify.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied by the new session policy if the rule criteria are met.

---

unset vpn sessionPolicy

Use this command to remove vpn sessionPolicy settings.Refer to the set vpn sessionPolicy command for meanings of the arguments.

Synopsys

unset vpn sessionPolicy <name> [-rule] [-action]

show vpn sessionPolicy

Displays a session policy.

Synopsys

show vpn sessionPolicy [<name>]

Arguments

name

Name of the session policy to display.

summary

fullValues

format

level

Outputs

rule

The new rule associated with the policy. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide.

action

The new vpn session action the policy is using.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

vpn stats

Sep 22, 2015

The following operations can be performed on "vpn stats":

show vpn stats

show vpn stats is an alias for stat vpn

Synopsys

show vpn stats - alias for 'stat vpn'

vpn trafficAction

Sep 22, 2015

The following operations can be performed on "vpn trafficAction":

add | rm | set | unset | show

add vpn trafficAction

Creates an action to be applied by a policy that matches the traffic being processed.

Synopsys

add vpn trafficAction <name> <qual> [-appTimeout <mins>] [(-SSO (ON | OFF) [-formSSOAction <string>]) | -wanscaler (ON | OFF)] [-fta (ON | OFF)] [-kcdAccount <string>] [-samlSSOProfile <string>]

Arguments

name

Name for the traffic action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after a traffic action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

qual

Protocol, either HTTP or TCP, to be used with the action. If you specify TCP, single sign-on cannot be configured.

Possible values: http, tcp

appTimeout

Maximum amount of time, in minutes, a user can stay logged on to the web application.

Minimum value: 1

Maximum value: 715827

SSO

Provide single sign-on to the web application.

Possible values: ON, OFF

formSSOAction

Name of the form-based single sign-on profile. Form-based single sign-on allows users to log on one time to all protected applications in your network, instead of requiring them to log on separately to access each one.

fta

Specify file type association, which is a list of file extensions that users are allowed to open.

Possible values: ON, OFF

wanscaler

Use the Repeater Plug-in to optimize network traffic.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "None"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

---

rm vpn trafficAction

Removes a previously created traffic policy action.

Synopsys

rm vpn trafficAction <name>

Arguments

name

Name of the traffic policy action to remove.

---

set vpn trafficAction

Modifies a traffic policy action to be applied by the policy if the rule criteria are met.

Synopsys

set vpn trafficAction <name> [-appTimeout <mins>] [-SSO (ON | OFF) | -wanscaler (ON | OFF)] [-formSSOAction <string>] [-fta (ON | OFF)] [-kcdAccount <string>] [-samlSSOProfile <string>]

Arguments

name

Name of the traffic policy action to modify.

appTimeout

Maximum amount of time, in minutes, a user can stay logged on to the web application.

Minimum value: 1

Maximum value: 715827

SSO

Provide single sign-on to the web application.

Possible values: ON, OFF

formSSOAction

Name of the form-based single sign-on profile. Form-based single sign-on allows users to log on one time to all protected applications in your network, instead of requiring them to log on separately to access each one.

fta

Specify file type association, which is a list of file extensions that users are allowed to open.

Possible values: ON, OFF

wanscaler

Use the Repeater Plug-in to optimize network traffic.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "None"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

---

unset vpn trafficAction

Use this command to remove vpn trafficAction settings.Refer to the set vpn trafficAction command for meanings of the arguments.

Synopsys

unset vpn trafficAction <name> [-wanscaler] [-kcdAccount]

---

show vpn trafficAction

Displays information about all the configured traffic actions, or displays detailed information about the specified traffic

action.

Synopsys

show vpn trafficAction [<name>]

Arguments

name

Name of the traffic policy action for which to display detailed information.

summary

fullValues

format

level

Outputs

qual

The protocol that is set with the action, e.g. http or tcp.

appTimeout

The application timeout

SSO

Whether or not Single Sign On is enabled.

formSSOAction

Name of the form-based single sign-on profile. Form-based single sign-on allows users to log on one time to all protected applications in your network, instead of requiring them to log on separately to access each one.

fta

Whether or not file-type association is enabled.

wanscaler

Use the Repeater Plug-in to optimize network traffic.

kcdAccount

Kerberos constrained delegation account name

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

stateflag

devno

count

vpn trafficPolicy

Sep 22, 2015

The following operations can be performed on "vpn trafficPolicy":

add | rm | set | unset | show

add vpn trafficPolicy

Creates a traffic policy. A traffic policy conditionally sets Access Gateway traffic characteristics at run time. For an intranet resource, for example, the traffic policy parameters define the destination IP address, destination port, amount of time a user can stay logged on to the application, and HTTP compression.

Synopsys

add vpn trafficPolicy <name> <rule> <action>

Arguments

name

Name for the traffic policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: '"<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to traffic that matches the policy.

rm vpn trafficPolicy

Removes an existing traffic policy from Access Gateway.

Synopsys

rm vpn trafficPolicy <name>

Arguments

name

Name of the traffic policy to remove.

set vpn trafficPolicy

Modifies the specified parameters of an existing traffic policy.

Synopsys

set vpn trafficPolicy <name> [-rule <expression>] [-action <string>]

Arguments

name

Name of the traffic policy to modify.

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "'<string of 255 characters>" + "<string of 245 characters>"'

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \\ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to traffic that matches the policy.

unset vpn trafficPolicy

Use this command to remove vpn trafficPolicy settings.Refer to the set vpn trafficPolicy command for meanings of the arguments.

Synopsys

unset vpn trafficPolicy <name> [-rule] [-action]

show vpn trafficPolicy

Displays information about all Access Gateway traffic policies, or detailed information about the specified policy.

Synopsys

show vpn trafficPolicy [<name>]

Arguments

name

Name of the traffic policy for which to display detailed information.

summary

fullValues

format

level

Outputs

rule

The rule used by the vpn traffic policy. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

action

The action to be performed when the rule is matched.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

vpn url

Sep 22, 2015

The following operations can be performed on "vpn url":

add | rm | set | unset | show

add vpn url

Creates a bookmark link to an external or internal resource that appears on the Access Interface, according to type, as a web site link or file share link.

Synopsys

add vpn url <urlName> <linkName> <actualURL> [-clientlessAccess (ON | OFF)] [-comment <string>]

Arguments

urlName

Name of the bookmark link.

linkName

Description of the bookmark link. The description appears in the Access Interface.

actualURL

Web address for the bookmark link.

clientlessAccess

If clientless access to the resource hosting the link is allowed, also use clientless access for the bookmarked web address in the Secure Client Access based session. Allows single sign-on and other HTTP processing on Access Gateway for HTTPS resources.

Possible values: ON, OFF

Default value: OFF

comment

Any comments associated with the bookmark link.

Example

add vpn url ggl search www.google.com.

---

rm vpn url

Removes a bookmark link to an internal resource that appears in the Access Interface.

Synopsys

rm vpn url <urlName>

Arguments

urlName

Name of the bookmark link to remove.

Example

rm vpn url ggl

---

set vpn url

Modifies the specified parameters of a bookmark link to an internal resource that appears in the Access Interface.

Synopsys

set vpn url <urlName> [-linkName <string>] [-actualURL <string>] [-clientlessAccess (ON | OFF)] [-comment <string>]

Arguments

urlName

Name of the bookmark link.

linkName

Description of the bookmark link. The description appears in the Access Interface.

actualURL

Web address for the bookmark link.

clientlessAccess

If clientless access to the resource hosting the link is allowed, also use clientless access for the bookmarked web address in the Secure Client Access based session. Allows single sign-on and other HTTP processing on Access Gateway for HTTPS resources.

Possible values: ON, OFF

Default value: OFF

comment

Any comments associated with the bookmark link.

Example

set vpn url wiurl -clientlessAccess on

unset vpn url

Use this command to remove vpn url settings.Refer to the set vpn url command for meanings of the arguments.

Synopsys

unset vpn url <urlName> [-clientlessAccess] [-comment]

show vpn url

Displays information about all the configured bookmark links to internal resources that appear in the Access Interface, or displays detailed information about the specified bookmark link.

Synopsys

show vpn url [<urlName>]

Arguments

urlName

Name of the bookmark link for which to display detailed information.

summary

fullValues

format

level

Outputs

linkName

Description of the bookmark link. The description appears in the Access Interface.

actualURL

Web address for the bookmark link.

clientlessAccess

Whether clientless access is enabled for the url in other modes or not.

comment

Comments associated with this virtual server.

devno

count

stateflag

vpn vserver

Sep 22, 2015

The following operations can be performed on "vpn vserver":

add | rm | set | unset | bind | unbind | enable | disable | show | stat | rename

add vpn vserver

Creates an Access Gateway virtual server to allow authenticated users to access intranet resources, such as XenApp, XenDesktop, and web servers.

Synopsys

add vpn vserver <name> <serviceType> (<IPAddress> [-range <positive_integer>]) <port> [-state (ENABLED | DISABLED)] [-authentication (ON | OFF)] [-doubleHop (ENABLED | DISABLED)] [-maxAAAUsers <positive_integer>] [-icaOnly (ON | OFF)] [-downStateFlush (ENABLED | DISABLED)] [-Listenpolicy <expression> [-Listenpriority <positive_integer>]] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-icmpVsrResponse (PASSIVE | ACTIVE)] [-netProfile <string>] [-cginfraHomePageRedirect (ENABLED | DISABLED)] [-maxLoginAttempts <positive_integer> [-failedLoginTimeout <positive_integer>]] [-deploymentType <deploymentType>]

Arguments

name

Name for the Access Gateway virtual server. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the virtual server is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my server" or 'my server').

serviceType

Protocol used by the Access Gateway virtual server.

Possible values: SSL

Default value: NSSVC_SSL

IPAddress

IPv4 or IPv6 address of the Access Gateway virtual server. Usually a public IP address. User devices send connection requests to this IP address.

port

TCP port on which the virtual server listens.

Minimum value: 1

state

State of the virtual server. If the virtual server is disabled, requests are not processed.

Possible values: ENABLED, DISABLED

Default value: ENABLED

authentication

Require authentication for users connecting to Access Gateway.

Possible values: ON, OFF

Default value: ON

doubleHop

Use the Access Gateway appliance in a double-hop configuration. A double-hop deployment provides an extra layer of security for the internal network by using three firewalls to divide the demilitarized zone (DMZ) into two stages. Such a deployment can have one appliance in the DMZ and one appliance in the secure network.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxAAAUsers

Maximum number of concurrent user sessions allowed on this virtual server. The actual number of users allowed to log on to this virtual server depends on the total number of user licenses.

icaOnly

User can log on in basic mode only, through either Citrix Receiver or a browser. Users are not allowed to connect by using the Access Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

downStateFlush

Close existing connections when the virtual server is marked DOWN, which means the server might have timed out. Disconnecting existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups. Enable this setting on servers whose connections can safely be closed when they are marked DOWN. Do not enable DOWN state flush

on servers that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Listenpolicy

String specifying the listen policy for the Access Gateway virtual server. Can be either a named expression or a default syntax expression. The Access Gateway virtual server processes only the traffic for which the expression evaluates to true.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server, the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Maximum value: 100

tcpProfileName

Name of the TCP profile to assign to this virtual server.

httpProfileName

Name of the HTTP profile to assign to this virtual server.

comment

Any comments associated with the virtual server.

appflowLog

Log AppFlow records that contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. Also log records that contain application-level information, such as HTTP web addresses, HTTP request methods and response status codes, server response time, and latency.

Possible values: ENABLED, DISABLED

Default value: DISABLED

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If this parameter is set to ACTIVE, respond only if the virtual server is available. With the PASSIVE setting, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

netProfile

The name of the network profile.

cginfraHomePageRedirect

When client requests for Sharefile resource and AGEE finds that the user is unauthenticated or the user-session has expired then, disabling this option will take the user to the originally requested Sharefile resource after authentication (instead of taking the user to the default vpn homepage)

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

Maximum value: 255

deploymentType

Example

The following example creates a VPN vserver named myvpnvip which supports SSL portocol and with AAA functionality enabled: vserver myvpnvip SSL 65.219.17.34 443 -

---

rm vpn vserver

Removes an Access Gateway virtual server. Policies that are bound to the virtual server are automatically unbound.

Synopsys

rm vpn vserver <name>@ ...

Arguments

name

Name of the virtual server to remove.

Example

rm vserver vpn_vip

---

set vpn vserver

Modifies the specified parameters of an Access Gateway virtual server.

Synopsys

set vpn vserver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-authentication (ON | OFF)] [-doubleHop (ENABLED | DISABLED)] [-icaOnly (ON | OFF)] [-maxAAAUsers <positive_integer>] [-downStateFlush (ENABLED | DISABLED)] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog (ENABLED | DISABLED)] [-icmpVsrResponse (PASSIVE | ACTIVE)] [-netProfile <string>] [-cginfraHomePageRedirect (ENABLED | DISABLED)] [-maxLoginAttempts <positive_integer>] [-failedLoginTimeout <positive_integer>]

Arguments

name

Name of the virtual server to modify.

IPAddress

IPv4 or IPv6 address of the Access Gateway virtual server. Usually a public IP address. User devices send connection requests to this IP address.

authentication

Require authentication for users connecting to Access Gateway.

Possible values: ON, OFF

Default value: ON

doubleHop

Use the Access Gateway appliance in a double-hop configuration. A double-hop deployment provides an extra layer of security for the internal network by using three firewalls to divide the demilitarized zone (DMZ) into two stages. Such a deployment can have one appliance in the DMZ and one appliance in the secure network.

Possible values: ENABLED, DISABLED

Default value: DISABLED

icaOnly

User can log on in basic mode only, through either Citrix Receiver or a browser. Users are not allowed to connect by using the Access Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

maxAAAUsers

Maximum number of concurrent user sessions allowed on this virtual server. The actual number of users allowed to log on to this virtual server depends on the total number of user licenses.

downStateFlush

Close existing connections when the virtual server is marked DOWN, which means the server might have timed out. Disconnecting existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups. Enable this setting on servers whose connections can safely be closed when they are marked DOWN. Do not enable DOWN state flush on servers that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Listenpolicy

String specifying the listen policy for the Access Gateway virtual server. Can be either a named expression or a default syntax expression. The Access Gateway virtual server processes only the traffic for which the expression evaluates to true.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server, the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Maximum value: 100

tcpProfileName

Name of the TCP profile to assign to this virtual server.

httpProfileName

Name of the HTTP profile to assign to this virtual server.

comment

Any comments associated with the virtual server.

appflowLog

Log AppFlow records that contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. Also log records that contain application-level information, such as HTTP web addresses, HTTP request methods and response status codes, server response time, and latency.

Possible values: ENABLED, DISABLED

Default value: DISABLED

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If this parameter is set to ACTIVE, respond only if the virtual server is available. With the PASSIVE setting, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: NS_VSR_PASSIVE

netProfile

The name of the network profile.

cginfraHomePageRedirect

When client requests for Sharefile resource and AGEE finds that the user is unauthenticated or the user-session has expired then, disabling this option will take the user to the originally requested Sharefile resource after authentication (instead of taking the user to the default vpn homepage)

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

Maximum value: 255

failedLoginTimeout

Failed Login timeout

Minimum value: 1

---

unset vpn vserver

Use this command to remove vpn vserver settings.Refer to the set vpn vserver command for meanings of the arguments.

Synopsys

unset vpn vserver <name> [-authentication] [-doubleHop] [-icaOnly] [-maxAAAUsers] [-downStateFlush] [-Listenpolicy] [-Listenpriority] [-tcpProfileName] [-httpProfileName] [-comment] [-appflowLog] [-icmpVsrResponse] [-netProfile] [-cginfraHomePageRedirect] [-maxLoginAttempts]

---

bind vpn vserver

Binds attributes to the specified Access Gateway virtual server.

Synopsys

bind vpn vserver <name> [-policy <string> [-priority <positive_integer>] [-secondary] [-groupExtraction] [-gotoPriorityExpression <expression>] [-type <type>]] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL>] [-appController <URL>] [-sharefile <string>]

Arguments

name

Name of the virtual server.

policy

Name of a policy to bind to the virtual server (for example, the name of an authentication, session, or endpoint analysis policy).

intranetApplication

Name of the application to bind to the virtual server. Intranet applications are used to enable access to selected applications located in the internal network. They are required for any user connecting with the Access Gateway Plug-in for Java.

nextHopServer

Name of the next hop server to bind to the virtual server.

urlName

Web address of the next hop virtual server to bind to the virtual server.

intranetIP

The network id for the range of intranet IP addresses or individual intranet ip to be bound to the vserver.

staServer

Web address of the Secure Ticket Authority (STA) server, in the following format: 'http(s)://FQDN/URLPATH'

appController

AppController server, in the format 'http(s)://IP/FQDN'

sharefile

Sharefile server, in the format 'IP:PORT / FQDN:PORT'

---

unbind vpn vserver

Unbinds the specified attributes from a virtual server.

Synopsys

unbind vpn vserver <name> [-policy <string> [-secondary] [-groupExtraction] [-type <type>]] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL>] [-appController <URL>] [-sharefile <string>]

Arguments

name

Name of the virtual server from which to unbind an attribute.

policy

Name of the policy to unbind from the virtual server.

intranetApplication

Name of intranet application to unbind from the virtual server.

nextHopServer

Name of the next hop server to remove.

urlName

Web address of the next hop virtual server to unbind.

intranetIP

The range of IP addresses to unbind from the virtual server.

staServer

Web address of the Secure Ticket Authority (STA) server to remove, in the following format: 'http(s)://FQDN/URLPATH'

appController

AppController server to be removed, in the format 'http(s)://IP/FQDN'

sharefile

Sharefile server to be removed, in the format 'IP:PORT / FQDN:PORT'

---

enable vpn vserver

Enables an Access Gateway virtual server. Note: Virtual servers, when added, are enabled by default.

Synopsys

enable vpn vserver <name>@

Arguments

name

Name of the virtual server to be enabled.

Example

```
enable vserver vpn1
```

---

disable vpn vserver

Disables an Access Gateway virtual server. The virtual server is taken out of service.

Synopsys

disable vpn vserver <name>@

Arguments

name

Name of the virtual server to be disabled. The Access Gateway still responds to ARP and/or ping requests for the IP address of the virtual server. You can enable the Access Gateway virtual server again at any time, because the virtual server is still configured.

Example

```
disable vserver lb_vip
```

---

show vpn vserver

Displays information about all the configured Access Gateway virtual servers, or displays detailed information about the specified Access Gateway virtual server.

Synopsys

show vpn vserver [<name>] show vpn vserver stats - alias for 'stat vpn vserver'

Arguments

name

Name of the Access Gateway virtual server for which to show detailed information.

summary

fullValues

format

level

Outputs

IPAddress

The IP address of the virtual server.

value

Indicates whether or not the certificate is bound or if SSL offload is disabled.

port

The virtual TCP port of the VPN vserver.

range

The range of vpn vserver IP addresses. The new range of vpn vservers will have IP addresses consecutively numbered, starting with the primary address specified with the <ipaddress> argument.

serviceType

The vpn vserver's protocol type, Currently the only possible value is SSL.

type

Bindpoint to which the policy is bound

state

The current state of the Virtual server, e.g. UP, DOWN, BUSY, etc.

status

Whether or not this vserver responds to ARPs and whether or not round-robin selection is temporarily in effect.

cacheType

Virtual server's cache type. The options are: TRANSPARENT, REVERSE and FORWARD.

redirect

The cache redirect policy.

The valid redirect policies are:

l. CACHE - Directs all requests to the cache.

2. POLICY - Applies cache redirection policy to determine whether the request should be directed to the cache or origin. This is the default setting.

3. ORIGIN - Directs all requests to the origin server.

precedence

This argument is used only when configuring content switching on the specified virtual server. This is applicable only

if both the URL and RULE-based policies have been configured on the same virtual server.

It specifies the type of policy (URL or RULE) that takes precedence on the content switching virtual server. The default setting is RULE.

l URL - In this case, the incoming request is matched against the URL-based policies before the rule-based policies.

l RULE - In this case, the incoming request is matched against the rule-based policies before the URL-based policies.

For all URL-based policies, the precedence hierarchy is:

1. Domain and exact URL

2. Domain, prefix and suffix

3. Domain and suffix

4. Domain and prefix

5. Domain only

6. Exact URL

7. Prefix and suffix

8. Suffix only

9. Prefix only

10. Default

redirectURL

The URL where traffic is redirected if the virtual server in system becomes unavailable. WARNING! Make sure that the domain you specify in the URL does not match the domain specified in the -d domainName argument of the ###add cs policy### command. If the same domain is specified in both arguments, the request will be continuously redirected to the same unavailable virtual server in the system. If so, the user may not get the requested content.

authentication

Indicates whether or not authentication is being applied to incoming users to the VPN.

doubleHop

Indicates whether double hop functionality is enabled or not.

icaOnly

Indicates whether or ica only license feature is enabled or not.

maxAAAUsers

The maximum number of concurrent users allowed to login into this vserver at a time.

curAAAUsers

The number of current users logged in to this vserver.

curTotalUsers

The total number of current users connected through this virtual server.

domain

The domain name of the server for which a service needs to be added. If the IP Address has been specified, the domain name does not need to be specified.

rule

The name of the rule, or expression, if any, that policy for the vpn server is to use. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide. The default rule is ns_true.

policyName

The name of the policy, if any, bound to the vpn vserver.NOTE: This attribute is deprecated.Replaced by Policy field

policy

The name of the policy, if any, bound to the vpn vserver.

serviceName

The name of the service, if any, to which the vserver policy is bound.

weight

Weight for this service, if any. This weight is used when the system performs load balancing, giving greater priority to a specific service. It is useful when the services bound to a virtual server are of different capacity.

cacheVserver

The name of the default target cache virtual server, if any, to which requests are redirected.

backupVServer

The name of the backup vpn virtual server for this vpn virtual server.

priority

The priority, if any, of the vpn vserver policy.

cltTimeout

The idle time, if any, in seconds after which the client connection is terminated.

soMethod

VPN client applications are allocated from a block of Intranet IP addresses.

That block may be exhausted after a certain number of connections. This switch specifies the

method used to determine whether or not a new connection will spillover, or exhaust, the allocated block of

Intranet IP addresses for that application. Possible values are CONNECTION or DYNAMICCONNECTION.

CONNECTION means that a static integer value is the hard limit for the spillover threshold. The spillover

threshold is described below. DYNAMICCONNECTION means that the spillover threshold is set according to

the maximum number of connections defined for the vpn vserver.

soThreshold

VPN client applications are allocated from a block of Intranet IP addresses.

That block may be exhausted after a certain number of connections.

The value of this option is number of client connections after which the Mapped IP address is used

as the client source IP address instead of an address from the allocated block of Intranet IP addresses.

soPersistence

Whether or not cookie-based site persistance is enabled for this VPN vserver. Possible values are 'ConnectionProxy', HTTPRedirect, or NONE

soPersistenceTimeOut

The timeout, if any, for cookie-based site persistance of this VPN vserver.

actType

intranetApplication

The intranet vpn application.

nextHopServer

The name of the next hop server bound to vpn vserver.

urlName

The intranet url.

intranetIP

The network id for the range of intranet IP addresses or individual intranet ip to be bound to the vserver.

netmask

The netmask of the intranet ip or range.

staServer

Configured Secure Ticketing Authority (STA) server.

staAuthID

Authority ID of the STA Server. Authority ID is used to match incoming STA Tickets in the SOCKS/CGP protocol with the right STA Server.

appController

Configured AppController server in Cloud Gateway deployment.

sharefile

Configured Sharefile server in Cloud Gateway deployment. Format IP:PORT / FQDN:PORT

useMIP

Deprecated. See 'map' below.

map

Whether or not Mapped IP Addresses are ON or OFF. Mapped IP addresses are source IP addresses

for the virtual servers running on the NetScaler. Mapped IP addresses are used by the system to connect to the backend servers.

downStateFlush

Close existing connections when the virtual server is marked DOWN, which means the server might have timed out. Disconnecting existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups. Enable this setting on servers whose connections can safely be closed when they are marked DOWN. Do not enable DOWN state flush on servers that must complete their transactions.

gotoPriorityExpression

Next priority expression.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

Listenpolicy

The string is listenpolicy configured for VPN vserver

Listenpriority

This parameter is the priority for listen policy of VPN Vserver.

tcpProfileName

Name of the TCP profile to assign to this virtual server.

httpProfileName

Name of the HTTP profile to assign to this virtual server.

policySubType

stateflag

flags

comment

Any comments associated with the virtual server.

appflowLog

Log AppFlow records that contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. Also log records that contain application-level information, such as HTTP web addresses, HTTP request methods and response status codes, server response time, and latency.

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If this parameter is set to ACTIVE, respond only if the virtual server is available. With the PASSIVE setting, respond even if the virtual server is not available.

netProfile

The name of the network profile.

cginfraHomePageRedirect

When client requests for Sharefile resource and AGEE finds that the user is unauthenticated or the user-session has expired then, disabling this option will take the user to the originally requested Sharefile resource after authentication (instead of taking the user to the default vpn homepage)

maxLoginAttempts

Maximum Number of login Attempts

failedLoginTimeout

Failed Login timeout

secondary

Bind the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only via a primary authentication method but also via a secondary authentication method. User groups are aggregated across both. The user name must be exactly the same for both authentication methods, but they can require different passwords.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

deploymentType

devno

count

Example

show vpn vserver

---

stat vpn vserver

Displays statistics for all Access Gateway virtual servers, or displays detailed statistics for the specified Access Gateway virtual server.

Synopsys

stat vpn vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the virtual server for which to show detailed statistics.

clearstats

Clear the statsistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

rename vpn vserver

Renames an Access Gateway virtual server.

Synopsys

rename vpn vserver <name>@ <newName>@

Arguments

name

Name of the Access Gateway virtual server.

newName

New name for the Access Gateway virtual server. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my server" or 'my server').

Example

```
rename vpn vserver vpn1 vpn1new
```

WebInterface Commands

Sep 22, 2015

The entities on which you can perform NetScaler CLI operations:

- wi package
- wi site

wi package

Sep 22, 2015

The following operations can be performed on "wi package":

install | uninstall

install wi package

Installs Web Interface and JRE tar files on the NetScaler appliance.

Synopsys

install wi package [-jre <URL>] [-wi <URL>] [-maxSites <maxSites>]

Arguments

jre

Complete path to the JRE tar file.

You can use the Diablo Latte JRE version 1.6.0-7 for 64-bit FreeBSD 6.x/amd64 platform available on the FreeBSD Foundation web site.

Alternatively, you can use OpenJDK6 package for FreeBSD 6.x/amd63.The Java package can be downloaded from https://citrix.sharefile.com/d/c85aeefcc05643f8 or http://www.freebsdfoundation.org/java/java16

Default value: "file://tmp/diablo-jdk-freebsd6.amd64.1.6.0.07.02.tbz"

wi

Complete path to the Web Interface tar file for installing the Web Interface on the NetScaler appliance. This file includes Apache Tomcat Web server. The file name has the following format: nswi-<version number>.tgz (for example, nswi-1.5.tgz).

Default value: "http://citrix.com/downloads/nswi-1.7.tgz"

maxSites

Maximum number of Web Interface sites that can be created on the NetScaler appliance; changes the amount of RAM reserved for Web Interface usage; changing its value results in restart of Tomcat server and invalidates any existing Web Interface sessions.

Possible values: 3, 25, 50, 100, 200, 500

Example

```
install wi package -jre http://10.102.1.10/diablo-latte-freebsd6-amd64-1.6.0_07-b02.tar.bz2 -wi http://citrix.com/downloads/nswi-1.7.tgz -maxSites 25
```

---

uninstall wi package

Removes the Web Interface and JRE tar files, and the entire Web Interface related configuration, from the NetScaler appliance.

Synopsys

uninstall wi package

Example

```
uninstall wi package
```

---

wi site

Sep 22, 2015

The following operations can be performed on "wi site":

add | rm | set | unset | bind | unbind | show

add wi site

Creates a Web Interface site on the NetScaler appliance. The NetScaler Web Interface feature provides access to Citrix XenApp and Citrix XenDesktop applications. Users access resources through a standard web browser or by using the Citrix XenApp plug-in.

Synopsys

add wi site <sitePath> [<agURL> [<staURL> [-secondSTAURL <string> [-useTwoTickets (ON | OFF)]] [-sessionReliability (ON | OFF)]] [-authenticationPoint (WebInterface | AccessGateway) [-agAuthenticationMethod (Explicit | SmartCard)]]] [-wiAuthenticationMethods (Explicit | Anonymous) ...] [-defaultCustomTextLocale <defaultCustomTextLocale>] [-webSessionTimeout <positive_integer>] [-defaultAccessMethod <defaultAccessMethod>] [-loginTitle <string>] [-appWelcomeMessage <string>] [-welcomeMessage <string>] [-footerText <string>] [-loginSysMessage <string>] [-preLoginButton <string>] [-preLoginMessage <string>] [-preLoginTitle <string>] [-domainSelection <string>] [-siteType (XenAppWeb | XenAppServices) [-ShowSearch (ON | OFF)] [-ShowRefresh (ON | OFF)] [-wiUserInterfaceModes (SIMPLE | ADVANCED)] [-UserInterfaceLayouts <UserInterfaceLayouts>]] [-userInterfaceBranding (Desktops | Applications)] [-publishedResourceType <publishedResourceType>] [-kioskMode (ON | OFF)] [-restrictDomains (ON | OFF)] [-loginDomains <string>] [-hideDomainField (ON | OFF)] [-agCallbackURL <string>]

Arguments

sitePath

Path to the Web Interface site being created on the NetScaler appliance.

agURL

Call back URL of the Gateway.

wiAuthenticationMethods

The method of authentication to be used at Web Interface

Default value: WI_EXPLICIT

defaultCustomTextLocale

Default language for the Web Interface site.

Possible values: German, English, Spanish, French, Japanese, Korean, Russian, Chinese_simplified, Chinese_traditional

Default value: LANG_EN

webSessionTimeout

Time-out, in minutes, for idle Web Interface browser sessions. If a client's session is idle for a time that exceeds the time-out value, the NetScaler appliance terminates the connection.

Default value: 20

Minimum value: 1

Maximum value: 1440

defaultAccessMethod

Default access method for clients accessing the Web Interface site.

Note: Before you configure an access method based on the client IP address, you must enable USIP mode on the Web Interface service to make the client?s IP address available with the Web Interface.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Note: In the NetScaler command line, mapping entries can be created by using the bind wi site command.

Possible values: Direct, Alternate, Translated, GatewayDirect, GatewayAlternate, GatewayTranslated

loginTitle

A custom login page title for the Web Interface site.

Default value: "Welcome to Web Interface on NetScaler"

appWelcomeMessage

Specifies localized text to appear at the top of the main content area of the Applications screen. LanguageCode is en, de, es, fr, ja, or any other supported language identifier.

welcomeMessage

Localized welcome message that appears on the welcome area of the login screen.

footerText

Localized text that appears in the footer area of all pages.

loginSysMessage

Localized text that appears at the bottom of the main content area of the login screen.

preLoginButton

Localized text that appears as the name of the pre-login message confirmation button.

preLoginMessage

Localized text that appears on the pre-login message page.

preLoginTitle

Localized text that appears as the title of the pre-login message page.

domainSelection

Domain names listed on the login screen for explicit authentication.

siteType

Type of access to the Web Interface site. Available settings function as follows:

* XenApp/XenDesktop web site - Configures the Web Interface site for access by a web browser.

* XenApp/XenDesktop services site - Configures the Web Interface site for access by the XenApp plug-in.

Possible values: XenAppWeb, XenAppServices

Default value: WI_XENAPPWEB

userInterfaceBranding

Specifies whether the site is focused towards users accessing applications or desktops. Setting the parameter to Desktops changes the functionality of the site to improve the experience for XenDesktop users. Citrix recommends using this setting for any deployment that includes XenDesktop.

Possible values: Desktops, Applications

Default value: WI_UIBRAND_APP

publishedResourceType

Method for accessing the published XenApp and XenDesktop resources.

Available settings function as follows:

* Online - Allows applications to be launched on the XenApp and XenDesktop servers.

* Offline - Allows streaming of applications to the client.

* DualMode - Allows both online and offline modes.

Possible values: Online, Offline, DualMode

Default value: WI_ONLINE

kioskMode

User settings do not persist from one session to another.

Possible values: ON, OFF

Default value: OFF

ShowSearch

Enables search option on XenApp websites

Possible values: ON, OFF

Default value: OFF

ShowRefresh

Provides the Refresh button on the applications screen.

Possible values: ON, OFF

Default value: OFF

wiUserInterfaceModes

Appearance of the login screen.

* Simple - Only the login fields for the selected authentication method are displayed.

* Advanced - Displays the navigation bar, which provides access to the pre-login messages and preferences screens.

Possible values: SIMPLE, ADVANCED

Default value: WI_SIMPLE

UserInterfaceLayouts

Specifies whether or not to use the compact user interface.

Possible values: AUTO, NORMAL, COMPACT

Default value: WI_AUTO

restrictDomains

The RestrictDomains setting is used to enable/disable domain restrictions. If domain restriction is enabled, the LoginDomains list is used for validating the login domain. It is applied to all the authentication methods except Anonymous for XenApp Web and XenApp Services sites

Possible values: ON, OFF

Default value: OFF

loginDomains

[List of NetBIOS domain names], Domain names to use for access restriction.

Only takes effect when used in conjunction with the RestrictDomains setting.

hideDomainField

The HideDomainField setting is used to control whether the domain field is displayed on the logon screen.

Possible values: ON, OFF

Default value: OFF

agCallbackURL

Callback AGURL to which Web Interface contacts.

Example

```
add wi site /Citrix/PNAgent -siteType XenAppServices
```

---

rm wi site

Removes a Web Interface site from the NetScaler appliance.

Synopsys

```
rm wi site <sitePath>
```

Arguments

sitePath

Path to the Web Interface site being created on the NetScaler appliance.

Example

```
rm wi site /Citrix/PNAgent
```

---

set wi site

Modifies the parameters of a Web Interface site configured on the NetScaler appliance.

Synopsys

```
set wi site <sitePath> [-agURL <string>] [-staURL <string>] [-sessionReliability ( ON | OFF )] [-useTwoTickets ( ON | OFF )] [-secondSTAURL <string>] [-wiAuthenticationMethods ( Explicit | Anonymous ) ...] [-defaultAccessMethod <defaultAccessMethod>] [-defaultCustomTextLocale <defaultCustomTextLocale>] [-webSessionTimeout <positive_integer>] [-loginTitle <string>] [-appWelcomeMessage <string>] [-welcomeMessage <string>] [-footerText <string>] [-loginSysMessage <string>] [-preLoginButton <string>] [-preLoginMessage <string>] [-preLoginTitle <string>] [-domainSelection <string>] [-userInterfaceBranding ( Desktops | Applications )] [-authenticationPoint ( WebInterface | AccessGateway )] [-agAuthenticationMethod ( Explicit | SmartCard )] [-publishedResourceType <publishedResourceType>] [-kioskMode ( ON | OFF )] [-ShowSearch ( ON | OFF )] [-ShowRefresh ( ON | OFF )] [-wiUserInterfaceModes ( SIMPLE | ADVANCED )] [-UserInterfaceLayouts <UserInterfaceLayouts>] [-restrictDomains ( ON | OFF )] [-loginDomains <string>] [-hideDomainField ( ON | OFF )] [-agCallbackURL <string>]
```

Arguments

sitePath

Path to the Web Interface site being created on the NetScaler appliance.

agURL

Call back URL of the Gateway.

staURL

URL of the Secure Ticket Authority (STA) server.

sessionReliability

Enable session reliability through Access Gateway.

Possible values: ON, OFF

Default value: OFF

useTwoTickets

Request tickets issued by two separate Secure Ticket Authorities (STA) when a resource is accessed.

Possible values: ON, OFF

Default value: OFF

secondSTAURL

URL of the second Secure Ticket Authority (STA) server.

wiAuthenticationMethods

The method of authentication to be used at Web Interface

Default value: WI_EXPLICIT

defaultAccessMethod

Default access method for clients accessing the Web Interface site.

Note: Before you configure an access method based on the client IP address, you must enable USIP mode on the Web Interface service to make the client?s IP address available with the Web Interface.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Note: In the NetScaler command line, mapping entries can be created by using the bind wi site command.

Possible values: Direct, Alternate, Translated, GatewayDirect, GatewayAlternate, GatewayTranslated

defaultCustomTextLocale

Default language for the Web Interface site.

Possible values: German, English, Spanish, French, Japanese, Korean, Russian, Chinese_simplified, Chinese_traditional

Default value: LANG_EN

webSessionTimeout

Time-out, in minutes, for idle Web Interface browser sessions. If a client's session is idle for a time that exceeds the time-out value, the NetScaler appliance terminates the connection.

Default value: 20

Minimum value: 1

Maximum value: 1440

loginTitle

A custom login page title for the Web Interface site.

Default value: "Welcome to Web Interface on NetScaler"

appWelcomeMessage

Specifies localized text to appear at the top of the main content area of the Applications screen. LanguageCode is en, de, es, fr, ja, or any other supported language identifier.

welcomeMessage

Localized welcome message that appears on the welcome area of the login screen.

footerText

Localized text that appears in the footer area of all pages.

loginSysMessage

Localized text that appears at the bottom of the main content area of the login screen.

preLoginButton

Localized text that appears as the name of the pre-login message confirmation button.

preLoginMessage

Localized text that appears on the pre-login message page.

preLoginTitle

Localized text that appears as the title of the pre-login message page.

domainSelection

Domain names listed on the login screen for explicit authentication.

userInterfaceBranding

Specifies whether the site is focused towards users accessing applications or desktops. Setting the parameter to Desktops changes the functionality of the site to improve the experience for XenDesktop users. Citrix

recommends using this setting for any deployment that includes XenDesktop.

Possible values: Desktops, Applications

Default value: WI_UIBRAND_APP

authenticationPoint

Authentication point for the Web Interface site.

Possible values: WebInterface, AccessGateway

agAuthenticationMethod

Method for authenticating a Web Interface site if you have specified Web Interface as the authentication point.

Available settings function as follows:

* Explicit - Users must provide a user name and password to log on to the Web Interface.

* Anonymous - Users can log on to the Web Interface without providing a user name and password. They have access to resources published for anonymous users.

Possible values: Explicit, SmartCard

publishedResourceType

Method for accessing the published XenApp and XenDesktop resources.

Available settings function as follows:

* Online - Allows applications to be launched on the XenApp and XenDesktop servers.

* Offline - Allows streaming of applications to the client.

* DualMode - Allows both online and offline modes.

Possible values: Online, Offline, DualMode

Default value: WI_ONLINE

kioskMode

User settings do not persist from one session to another.

Possible values: ON, OFF

Default value: OFF

ShowSearch

Enables search option on XenApp websites

Possible values: ON, OFF

Default value: OFF

ShowRefresh

Provides the Refresh button on the applications screen.

Possible values: ON, OFF

Default value: OFF

wiUserInterfaceModes

Appearance of the login screen.

* Simple - Only the login fields for the selected authentication method are displayed.

* Advanced - Displays the navigation bar, which provides access to the pre-login messages and preferences screens.

Possible values: SIMPLE, ADVANCED

Default value: WI_SIMPLE

UserInterfaceLayouts

Specifies whether or not to use the compact user interface.

Possible values: AUTO, NORMAL, COMPACT

Default value: WI_AUTO

restrictDomains

The RestrictDomains setting is used to enable/disable domain restrictions. If domain restriction is enabled, the LoginDomains list is used for validating the login domain. It is applied to all the authentication methods except Anonymous for XenApp Web and XenApp Services sites

Possible values: ON, OFF

Default value: OFF

loginDomains

[List of NetBIOS domain names], Domain names to use for access restriction.

Only takes effect when used in conjunction with the RestrictDomains setting.

hideDomainField

The HideDomainField setting is used to control whether the domain field is displayed on the logon screen.

Possible values: ON, OFF

Default value: OFF

agCallbackURL

Callback AGURL to which Web Interface contacts.

Example

```
set wi site /Citrix/PNAgent -staURL http://myStaServer
```

---

unset wi site

Use this command to remove wi site settings.Refer to the set wi site command for meanings of the arguments.

Synopsys

```
unset wi site <sitePath> [-appWelcomeMessage] [-welcomeMessage] [-footerText] [-loginSysMessage] [-preLoginButton] [-preLoginMessage] [-preLoginTitle] [-userInterfaceBranding] [-loginDomains]
```

---

bind wi site

Binds XenApp or XenDesktop farms to a Web Interface site and optionally, defines access methods for different client IP addresses or networks.

Synopsys

```
bind wi site <sitePath> ((<farmName> <xmlServerAddresses> [-groups <string>] [-recoveryFarm ( ON | OFF )] [-xmlPort <positive_integer>] [-transport <transport> [-sslRelayPort <positive_integer>]] [-loadBalance ( ON | OFF )]) | ((-accessMethod <accessMethod> (-clientIpAddress <ip_addr> -clientNetMask <netmask>)) | (-translationInternalIp <ip_addr> -translationInternalPort <port | *> -translationExternalIp <ip_addr> -translationExternalPort <port | *> [-accessType <accessType>])))
```

Arguments

sitePath

Path to the Web Interface site.

farmName

Name for the logical representation of a XenApp or XenDesktop farm to be bound to the Web Interface site. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

accessMethod

Secure access method to be applied to the IPv4 or network address of the client specified by the Client IP Address parameter.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Possible values: Direct, Alternate, Translated, GatewayDirect, GatewayAlternate, GatewayTranslated

translationInternalIp

IP address of the server for which you want to associate an external IP address. (Clients access the server through the associated external address and port.)

Example

```
bind wi site /Citrix/XenApp Farm2 10.10.10.11
```

---

unbind wi site

Unbinds XenApp or XenDesktop farms from the Web Interface site and removes the existing access method definition for a client IP address or network.

Synopsys

```
unbind wi site <sitePath> (<farmName> | ((-clientIpAddress <ip_addr> -clientNetMask <netmask>) | (-translationInternalIp <ip_addr> -translationInternalPort <port | *> -translationExternalIp <ip_addr> -translationExternalPort <port | *>)))
```

Arguments

sitePath

Path to the Web Interface site.

farmName

Name of the XenApp farm to be unbound from the Web Interface site.

clientIpAddress

IPv4 address or network address of the client for which you want to remove the defined access method.

translationInternalIp

Internal IP address of a mapping entry to be removed.

Example

```
unbind wi site  /Citrix/XenApp Farm2
```

---

show wi site

Displays settings of all the Web Interface sites, or of a specified site. To display settings of all the Web Interface sites, run the command without any parameters.

Synopsys

```
show wi site [<sitePath>]
```

Arguments

sitePath

Path of a Web Interface site whose details you want the NetScaler appliance to display.

format

level

Outputs

stateflag

agURL

Call back URL of the Gateway.

staURL

The URL of Secure Ticketing Authortity server

wiAuthenticationMethods

The method of authentication to be used at Web Interface

loginTitle

A custom login page title for the Web Interface site.

appWelcomeMessage

Specifies localized text to appear at the top of the main content area of the Applications screen. LanguageCode is en, de, es, fr, ja, or any other supported language identifier.

welcomeMessage

Localized welcome message that appears on the welcome area of the login screen.

footerText

Localized text that appears in the footer area of all pages.

loginSysMessage

Localized text that appears at the bottom of the main content area of the login screen.

preLoginButton

Localized text that appears as the name of the pre-login message confirmation button.

preLoginMessage

Localized text that appears on the pre-login message page.

preLoginTitle

Localized text that appears as the title of the pre-login message page.

domainSelection

Domain names listed on the login screen for explicit authentication.

defaultCustomTextLocale

Default language for the Web Interface site.

webSessionTimeout

Time-out, in minutes, for idle Web Interface browser sessions. If a client's session is idle for a time that exceeds the time-out value, the NetScaler appliance terminates the connection.

siteType

Type of access to the Web Interface site. Available settings function as follows:

* XenApp/XenDesktop web site - Configures the Web Interface site for access by a web browser.

* XenApp/XenDesktop services site - Configures the Web Interface site for access by the XenApp plug-in.

userInterfaceBranding

Specifies whether the site is focused towards users accessing applications or desktops. Setting the parameter to Desktops changes the functionality of the site to improve the experience for XenDesktop users. Citrix recommends using this setting for any deployment that includes XenDesktop.

ShowSearch

Enables search option on XenApp websites

ShowRefresh

Provides the Refresh button on the applications screen.

wiUserInterfaceModes

Appearance of the login screen.

* Simple - Only the login fields for the selected authentication method are displayed.

* Advanced - Displays the navigation bar, which provides access to the pre-login messages and preferences screens.

UserInterfaceLayouts

Specifies whether or not to use the compact user interface.

publishedResourceType

Method for accessing the published XenApp and XenDesktop resources.

Available settings function as follows:

* Online - Allows applications to be launched on the XenApp and XenDesktop servers.

* Offline - Allows streaming of applications to the client.

* DualMode - Allows both online and offline modes.

defaultAccessMethod

Default access method for clients accessing the Web Interface site.

Note: Before you configure an access method based on the client IP address, you must enable USIP mode on the Web Interface service to make the client?s IP address available with the Web Interface.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Note: In the NetScaler command line, mapping entries can be created by using the bind wi site command.

farmName

Name for the logical representation of a XenApp or XenDesktop farm to be bound to the Web Interface site. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

accessMethod

Secure access method to be applied to the IPv4 or network address of the client specified by the Client IP Address parameter.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

clientIpAddress

IPv4 or network address of the client for which you want to associate an access method.

clientNetMask

Subnet mask associated with the IPv4 or network address specified by the Client IP Address parameter.

translationInternalIp

IP address of the server for which you want to associate an external IP address. (Clients access the server through the associated external address and port.)

translationInternalPort

Port number of the server for which you want to associate an external port. (Clients access the server through the associated external address and port.)

translationExternalIp

External IP address associated with server's IP address.

translationExternalPort

External port number associated with the server's port number.

accessType

Type of access to the XenApp or XenDesktop server.

Available settings function as follows:

* User Device - Clients can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

* Gateway - Access Gateway can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

* User Device and Gateway - Both clients and Access Gateway can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

xmlServerAddresses

Comma-separated IP addresses or host names of XenApp or XenDesktop servers providing XML services.

xmlPort

Port number at which to contact the XML service.

transport

Transport protocol to use for transferring data, related to the Web Interface site, between the NetScaler appliance and the XML service.

sslRelayPort

TCP port at which the XenApp or XenDesktop servers listenfor SSL Relay traffic from the NetScaler appliance. This parameter is required if you have set SSL Relay as the transport protocol.

Web Interface uses root certificates when authenticating a server running SSL Relay. Make sure that all the servers running SSL Relay are configured to listen on the same port.

agAuthenticationMethod

Method for authenticating a Web Interface site if you have specified Web Interface as the authentication point.

Available settings function as follows:

* Explicit - Users must provide a user name and password to log on to the Web Interface.

* Anonymous - Users can log on to the Web Interface without providing a user name and password. They have access to resources published for anonymous users.

groups

Active Directory groups that are permitted to enumerate resources from server farms. Including a setting for this parameter activates the user roaming feature. A maximum of 512 user groups can be specified for each farm defined with the Farm<n> parameter. The groups must be comma separated.

recoveryFarm

Binded farm is set as a recovery farm.

sessionReliability

Enable session reliability through Access Gateway.

useTwoTickets

Request tickets issued by two separate Secure Ticket Authorities (STA) when a resource is accessed.

secondSTAURL

URL of the second Secure Ticket Authority (STA) server.

loadBalance

Use all the XML servers (load balancing mode) or only one server (failover mode).

authenticationPoint

Authentication point for the Web Interface site.

kioskMode

User settings do not persist from one session to another.

restrictDomains

The RestrictDomains setting is used to enable/disable domain restrictions. If domain restriction is enabled, the LoginDomains list is used for validating the login domain. It is applied to all the authentication methods except Anonymous for XenApp Web and XenApp Services sites

loginDomains

[List of NetBIOS domain names], Domain names to use for access restriction.

Only takes effect when used in conjunction with the RestrictDomains setting.

hideDomainField

The HideDomainField setting is used to control whether the domain field is displayed on the logon screen.

agCallbackURL

Callback AGURL to which Web Interface contacts.

devno

count

Example

show wi site

Quick Start Guides

Oct 23, 2013

A reference to quick installation and configuration of your hardware appliance.

Title
Citrix NetScaler Quick Start Guide for MPX 5500
Citrix NetScaler Quick Start Guide for MPX 5550, 5650
Citrix NetScaler Quick Start Guide for MPX 7500, 9500
Citrix NetScaler Quick Start Guide for MPX 8200, 8400, 8600, 8800
Citrix NetScaler Quick Start Guide for MPX 9700, 10500, 12500, 15500
Citrix NetScaler Quick Start Guide for MPX 11500, 13500, 14500, 16500, 18500, 20500
Citrix NetScaler Quick Start Guide for MPX 15000, 17000
Citrix NetScaler Quick Start Guide for MPX 17500, 19500, 21500
Citrix NetScaler Quick Start Guide for MPX 17550, 19550, 20550, 21550
Citrix NetScaler Quick Start Guide for MPX 22040, 22060, 22080, 22100, 22120
Citrix NetScaler Quick Start Guide for MPX 24100, 24150
Citrix NetScaler Quick Start Guide for MPX 25100T, 25160T

Glossary

Jan 28, 2011

A

AAA

A NetScaler feature providing authentication, authorization, and auditing for all application traffic.

AAA-TM

See AAA. The configuration utility displays AAA as AAA-TM, meaning AAA traffic management.

access control

A general term denoting something that controls access to a resource. A more specific term is usually preferable.

Access Gateway

Former name of NetScaler Gateway.

action

A policy element that specifies what to do with a request or response that matches the expression in the policy. For example, if an expression in a policy matches a particular source IP address in a request, the action associated with the policy determines whether the connection is permitted.

action analytics

A NetScaler data-collection feature that can automatically optimize traffic in real time.

active-active mode

A deployment mode that, in addition to preventing downtime, makes efficient use of all the NetScaler ADCs in the deployment. In active-active deployment mode, the same virtual IP (VIP) addresses are assigned to all NetScaler ADCs in the configuration, but with different priorities, so that a given VIP can be active on only one ADC at a time. The ADCs can be configured so that no NetScaler ADC is idle.

ADC

See application delivery controller.

Amazon Elastic Block Store (EBS)

AWS feature that provides storage volumes that can be attached to EC2 instances.

Amazon Machine Image (AMI)

A special type of virtual appliance used to instantiate (create) a virtual machine within the Amazon Elastic Compute Cloud (EC2). It serves as the basic unit of deployment for services delivered through EC2.

AMI

See Amazon Machine Image.

AppFlow

A NetScaler feature that provides transaction-level visibility into HTTP, SSL, TCP, and SSL_TCP traffic flows.

application delivery controller (ADC)

A product, such as Citrix NetScaler, that optimizes delivery of applications. An ADC provides advanced features in addition to basic load balancing.

application visualizer

A graphical representation of an AppExpert application. Displays the public endpoints, application units, backend services, and policies that are configured for the application. You can use the Visualizer to obtain a visual overview of an AppExpert application's configuration and configure some of the displayed entities. By default, the Visualizer displays application units, services, and monitors for the selected application.

AVP

See Attribute-Value Pair.

Attribute-Value Pair (AVP)

AVPs are the basic units inside a Diameter and Radius message that carry authentication, security, and any other data pertaining to the application. There must be at least one AVP inside a Diameter or RADIUS message.

auditing

Feature that keeps a record of each user's activity on a protected server.

authentication

Feature for verifying client credentials, either locally or with a third-party authentication server, and allowing only approved users to access protected servers.

authorization

Feature for verifying which content on a protected server each user is allowed to access.

AWS region

Amazon EC2 is hosted in multiple locations world-wide. These locations are composed of regions and Availability Zones. Each region is a separate geographic area. Each region has multiple, isolated locations known as Availability Zones. Amazon EC2 gives you the ability to place resources, such as instances, and data in multiple locations. Resources are not replicated across regions unless you do so specifically.

B

back end

The server-facing side of a network.

bind point

An entity, or a stage of traffic processing, at which traffic is examined to see if it matches a policy. For example, a bind point can be a load balancing virtual server, or it can apply to all traffic at given stage of processing, such as when a request is received or when a response is sent.

bridge group

A NetScaler feature for merging multiple VLANs into a single broadcast domain.

C

cache redirection

A policy and virtual-server based NetScaler feature that evaluates the type of content requested and directs requests to a cache instead of a server.

call home

A NetScaler feature that monitors the appliance and automatically uploads data to the Support server if an error condition is detected.

CEA

Capabilities Exchange Answer. A message used by the Diameter protocol to establish a connection.

CER

Capabilities Exchange Request. A message used by the Diameter protocol to establish a connection.

certificate-key pair

An SSL certificate and its corresponding private key. Stored on a NetScaler ADC that offloads SSL processing from servers.

classic policy

The older, less robust type of NetScaler policy.

CLI

Command-line interface.

client

A computer that receives data from a server. Can also refer to a person.

client data plane

The logical grouping of the physical connections between the cluster nodes and the client-side connecting device.

client drive mapping

A method that enables users to access some or all of their clients' drives from an application running on an application server.

client keep-alive

A setting that enables receiving multiple client requests on a single client connection. Applies only to HTTP and HTTPS services.

CloudBridge Connector

A NetScaler feature for connecting a datacenter to a cloud or another datacenter. The feature establishes a "CloudBridge Connector tunnel" between the connected entities.

CloudFormation

An Amazon Web Services (AWS) feature for managing cloud resources. Uses templates to manage groups of resources, which are called "stacks."

CloudPlatform

A Citrix software platform (powered by Apache CloudStack) that pools computing resources to build public, private, and hybrid Infrastructure as a Service (IaaS) clouds.

cluster

A group of nCore appliances working together as a single system image. The cluster can include as few as 2 or as many as 32 NetScaler nCore hardware or virtual appliances as nodes. The client traffic is distributed between the nodes to provide high availability, high throughput, and scalability.

cluster backplane

The logical grouping of the physical connections between the cluster nodes and the cluster backplane switch. The nodes of a cluster communicate with each other over the cluster backplane.

cluster backplane switch

A switch through which cluster nodes communicate with each other.

cluster instance

A logical entity created on the first node added to a NetScaler cluster. The cluster instance is assigned a cluster ID, which uniquely identifies the cluster.

cluster IP address

The management IP address of a cluster, through which configuration tasks must be performed. This IP address is owned by the cluster's configuration coordinator.

cluster link aggregation

A feature combining groups of cluster interfaces into channels. Similar to NetScaler link aggregation, but without the requirement that all interfaces be on the same appliance. The interfaces can be on different nodes of the same cluster.

cluster node

A NetScaler ADC that is part of a cluster.

cluster propagation

The process through which configurations that are performed on the cluster IP address are propagated to all the nodes of the cluster.

cluster synchronization

The process through which cluster configurations are synchronized to appliances that are added as cluster nodes or to nodes that rejoin the cluster.

clustering

The process of creating a cluster of NetScaler ADCs.

collector

An AppFlow entity that receives flow records generated by a NetScaler appliance.

community string

A password for authenticating SNMP queries from SNMP managers.

configuration coordinator

The cluster node on which cluster configurations are performed and then propagated to the other cluster nodes. The configuration coordinator owns the cluster IP address.

configuration utility

The NetScaler graphical user interface (GUI).

console

The command line interface accessed through the console port of a NetScaler appliance.

content filtering

A NetScaler feature that takes a user-specified action when the something in the header of a request or response matches a policy.

content group

A group of all virtual servers and policies involved in a particular content switching configuration.

content switching

Load balancing that bases server selection on the type of content requested.

critical interface

A NetScaler interface that, if it fails or is disabled, triggers a high-availability (HA) failover.

crossover cable

An Ethernet cable in which the sending and receiving wires are crossed.

D

dashboard

An interface element that displays performance data in graphical form.

data set

A specialized form of pattern set, consisting of an array of patterns of type number (integer), IPv4 address, or IPv6 address.

datacenter

A facility housing computer systems and associated components, such as telecommunications and storage systems. Usually includes redundant or backup power supplies, redundant data communications connections, environmental controls such as air conditioning and fire suppression, and security devices.

DataStream

NetScaler feature for load balancing database servers.

default syntax policies

The newer type of NetScaler policies, which provide more capabilities than do classic policies. Most NetScaler features are migrating from classic to default syntax policies.

Desktop Director

A Citrix product that provides a detailed and intuitive overview of XenDesktop environments.

Diameter

An Authentication, Authorization, and Accounting (AAA) protocol derived from RADIUS.

direct server return (DSR)

A NetScaler load balancing mode in which servers send responses directly to the clients, instead of through the NetScaler ADC.

Disconnect Peer Acknowledgment (DPA)

A response acknowledging a DPR.

Disconnect Peer Request (DPR)

A Diameter request sent to a peer to initiate session termination.

down state flush

A NetScaler feature for delayed cleanup of a virtual-server's connections. Connections remain open until the virtual server enters the DOWN state.

DPA

See Disconnect Peer Acknowledgment.

DPR

See Disconnect Peer Request.

DSR

See direct server return.

E

EBS

See Amazon Elastic Block Store.

EC2

See Elastic Cloud Compute.

effective state

Cumulative state of the primary and backup virtual servers. If any of virtual servers in the chain is UP, the effective state is UP. For a GSLB service, the effective state reflects the effective state of the corresponding load balancing virtual server. (A load balancing *service* has no effective state.)

Elastic Cloud Compute (EC2)

Amazon Web Services (AWS) feature with which users create virtual computers (instances).

Elastic Network Interface (ENI)

An Amazon Web Services (AWS) virtual network interface that you can attach to an instance in a virtual private cloud (VPC).

ENI

See Elastic Network Interface.

ETag

An identifier assigned by a web server to a specific resource at a URL. Useful for cache validation and for preventing one simultaneous update of a resource from overwriting another.

expression

A logic statement, such as a Perl Compatible Regular Expression (PCRE), specifying the characteristics of requests or responses that match the policy of which the expression is a part. Also called a rule.

F

failover interface set

A pair of interfaces, with each interface on a different appliance. If one appliance fails, process is transferred to the other appliance without triggering a failover event.

Federal Information Processing Standards (FIPS)

Standards developed by the National Institute for Standards and Technology (NIST) to ensure compliance with federal security and data-privacy requirements.

field replaceable unit (FRU)

A NetScaler component that can be replaced by the user.

FIPS

See Federal Information Processing Standards.

FRU

See field replaceable unit.

flow processor

The cluster node that is selected as the node to process the traffic. The flow processor receives the traffic from the flow receiver through the cluster backplane.

flow receiver

The cluster node that receives traffic from the external network. The flow receiver node steers or forwards the traffic to the flow processor through the cluster backplane.

front end

The client-facing side of a network.

G

global server load balancing (GSLB)

A NetScaler feature that performs load balancing across data centers in a WAN.

GSLB

See global server load balancing.

GUI

See configuration utility.

H

HDX Insight

NetScaler Insight Center component that monitors ICA traffic.

high availability (HA)

A deployment mode in which one appliance (the primary) is backed up by another appliance (the secondary). If the primary appliance fails, a failover event transfers control to the secondary appliance.

HA

See high availability.

I

IAM

See Identity and Access Management.

INC

See Independent Network Configuration (INC) mode.

Identity and Access Management (IAM)

An Amazon Web Services (AWS) feature with which you can control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups and use permissions to allow or deny their access to AWS resources.

ICA

See Independent Computing Architecture (ICA)

Independent Computing Architecture (ICA)

Citrix proprietary protocol for XenApp and XenDesktop traffic.

Independent Network Configuration (INC) mode

A type of High availability deployment in which the two HA nodes reside in different networks. The following independent network entities and configurations are neither propagated nor synced to the other node: MIPs, SNIPs, VLANs, routes (except LLB routes), route monitors, RNAT rules (except any RNAT rule with a VIP as the NAT IP), and dynamic routing configurations.

information element

A description of an attribute that can appear in an IPFIX Record. RFC5102 defines the base set of IPFIX information elements.

inline mode

A two-arm deployment mode in which the traffic between clients and servers passes through the deployed appliance.

IP set

A set of subnet IP (SNIP) addresses and virtual IP (VIP) addresses, identified with a meaningful name indicating the usage of the IP addresses contained in the set.

L

link load balancing (LLB)

A NetScaler feature that balances outbound traffic across multiple Internet connections provided by different service providers.

Linkset

An entity specifying interfaces through which a node can connect to the external switch through the cluster backplane. Linksets must be used for traffic distribution in an asymmetric (some nodes not connected to the external switch) cluster topology. Linksets can be used exclusively or combined with ECMP or cluster link aggregation.

load balancing

A core NetScaler feature that distributes user requests for web pages and other protected applications across multiple

servers that all host (or mirror) the same content.

load balancing virtual server

The IP address, port, and protocol combination to which a client sends connection requests for a particular load-balanced website or application. If the application is accessible from the Internet, the virtual server IP (VIP) address is a public IP address. If the application is accessible only from the local area network (LAN) or wide area network (WAN), the VIP is usually a private (ICANN non-routable) IP address.

M

management service

The graphical user interface (GUI) of a NetScaler SDX appliance. Also used on some CloudBridge appliances.

mapped IP (MIP)

Mapped IP address. A NetScaler IP address used for server-side connections. Citrix recommends using a subnet IP (SNIP) address instead.

MEP

Metric Exchange Protocol, a Citrix protocol used for GSLB.

MIP

See mapped IP.

MIR

See multiple IP response.

monitor

A NetScaler entity that periodically probes each service to which you assign it. If a service does not respond within a specified interval, and the specified number of probes fail, the service is marked DOWN. In that case, the monitor continues to send probes, and can change the status to UP.

multiple IP response (MIR)

A GSLB option for the NetScaler ADC to send multiple IP addresses in response to a DNS request.

multitenant

A configuration in which multiple clients are served from the same platform.

N

named expression

An expression that has been assigned a name, which is used instead of the expression itself in a policy.

nCore

The multiple-core, 64-bit version of the NetScaler operating system.

negative caching

The caching of negative responses from servers in a domain, to speed up responses to queries.

net profile

An information set that contains NetScaler owned IP addresses (IP set) or an IP address. During communication with

physical servers or peers, the NetScaler ADC uses the addresses specified in the profile as source IP addresses.

netbridge

A logical container that holds or represents a CloudBridge Connector tunnel configuration on a NetScaler appliance. A GRE tunnel entity is associated with the netbridge. A particular CloudBridge Connector tunnel configuration on a NetScaler appliance is identified by the name of the netbridge entity.

netmask

A network mask. Also called a subnet mask.

NetScaler Gateway

A NetScaler feature (also available as a standalone appliance) that provides secure access to a LAN or WAN from any location on the Internet.

NetScaler Insight Center

A Citrix virtual appliance that can monitor NetScaler appliances.

NetScaler IP (NSIP)

NetScaler IP address. The IP address for management and general system access to the NetScaler appliance.

NetScaler owned IP address

An IP address that exists only on a NetScaler ADC. NetScaler owned IP addresses can be of the following types: NetScaler IP address (NSIP), Virtual IP addresses (VIPs), Subnet IP addresses (SNIPs), and global server load balancing site IP addresses (GSLBIPs). The NetScaler IP address (NSIP) uniquely identifies the NetScaler ADC on your network, and it provides access to the ADC. A Virtual IP address (VIP) is a public IP address to which a client sends requests. The NetScaler ADC terminates the client connection at the VIP and initiates a connection with a server. This new connection uses a subnet IP address (SNIP) as the source IP address for packets forwarded to the server. If you have multiple data centers that are geographically distributed, each data center can be identified by a unique global server load balancing site IP address (GSLBIP).

NetScaler software

The NetScaler operating system.

NetScaler VPX

A NetScaler virtual machine image that can be installed on a virtualization platform.

NetScaler Web Logging (NSWL) client

Software installed on the client system to collect logs of HTTP and HTTPS requests.

network visualizer

A Network feature that displays the network configuration of a NetScaler ADC, including the network configuration of the nodes in a high availability (HA) deployment. You can also modify the configuration of VLANs, interfaces, channels, and bridge groups, and perform HA configuration tasks.

Next Secure (NSEC)

A DNS record showing that no records exist between two points.

NITRO

The NetScaler API suite.

node group

A group of cluster nodes that have a specific set of cluster configurations. Node groups are used to define partially striped configurations.

node instance

A logical entity on a cluster node. The node instance is assigned a node ID, which uniquely identifies the node.

non-INC mode

A high availability-deployment mode in which both HA nodes are in the same network.

NSIP

See NetScaler IP.

NSVLAN

The virtual LAN (VLAN) to which the subnet that includes the NetScaler management IP (NSIP) address is bound. This subnet is available only on interfaces that are associated with NSVLAN.

O

one-arm mode

Configuration in which only one NetScaler interface is connected to an Ethernet segment.

P

partially striped configuration

A configuration available on a subset of cluster nodes. The subset is defined by the node group.

pattern set

An array of indexed patterns used for string matching during default syntax policy evaluation. Example of a pattern set: image types {svg, bmp, png, gif, tiff, jpg}. Also called a patset.

policy

An entity that identifies requests or responses on which to perform specified actions. A policy is essentially of the form: if <expression>, do <action>

policy binding

The act of binding a policy to a bind point, which determines the instant at which the policy is invoked. A policy can be bound to a virtual server or globally to the NetScaler appliance.

pre-shared key

A text string manually configured on CloudBridge peers. The strings are matched against each other for authentication before security associations are established.

profile

A collection of settings that enable a feature to perform a complex function. For example, in the application firewall, a profile for XML data can perform multiple screening operations, such as examining the data for illegal XML syntax or evidence of SQL injection.

R

rate limit identifier

A named entity that specifies numeric rate-limiting thresholds, such as the maximum number of requests or connections (of a particular type) that are permitted in a specified period called a *time slice*.

rate limiting

A NetScaler feature with which you can configure the appliance to monitor the rate of traffic associated with an entity and take preventive action, in real time, when the rate reaches a specified value.

reboot

To restart an appliance.

redirection

Sending a client request to a different web page or server.

Request Switching

Citrix technology that enables an appliance to multiplex and offload TCP connections, maintain persistent connections, and manage traffic at the request (application layer) level.

responder

NetScaler feature that sends an automatic response based on who sent the request, where it is from, and other criteria with security and system-management implications.

REST interface

REpresentational State Transfer interface. A software architecture for using simple HTML calls to create or modify information on a server.

rewrite

NetScaler feature that modifies information in the headers or bodies of requests or responses.

rule

A policy element consisting of a logical expression used to evaluate requests or responses. If the evaluation returns TRUE, the action that is bound to the policy is performed.

S

SDX

An advanced NetScaler or CloudBridge platform hosting virtual machines (VMs). NetScaler SDX hosts multiple NetScaler VMs. CloudBridge SDX hosts a NetScaler VM and multiple CloudBridge VMs.

Secure Ticket Authority (STA)

The XenApp/XenDesktop entity responsible for issuing session tickets in response to connection requests for published applications on XenApp and published desktops on XenDesktop. These session tickets form the basis of authentication and authorization for access to published resources.

selectlets

A group of non-compound, default syntax expressions, each of which is called a selectlet. A traffic stream selector can contain up to five selectlets. Each selectlet is considered to be in an AND relationship with the other expressions.

selector

A filter for identifying requests, or for identifying objects in a content group.

server data plane

The logical grouping of the physical connections between cluster nodes and the server-side connecting device.

server object

A virtual entity representing a physical server. Enables naming a server, rather than identifying it by its IP address.

service

The IP address, port, and protocol combination used to route requests to a specific load-balanced application server. A service can be a logical representation of the application server itself, or of an application running on a server that hosts multiple applications. After creating a service, you bind it to a virtual server.

sessionless load balancing

Load balancing on a per packet basis, without storing session information. Reduces NetScaler resource requirements. Used in DSR mode.

shell

Refers to the BSD command shell unless otherwise stated.

single sign-on (SSO)

A method of providing access to multiple password-protected information systems without requiring multiple authentications. Supported by the Citrix password manager.

single-hop mode

The mode in which NetScaler Insight Center collects data from NetScaler appliances handling connections in which users connect to XenApp and XenDesktop applications through a NetScaler Gateway appliance.

slow start

A NetScaler feature that avoids assigning all new connections to a server when it is added to the network.

spotted configuration

A configuration that is available on only a single cluster node.

SSO profile

In forms-based single signon (SSO), the profile that defines how to handle an authentication request that matches the associated policy.

start

To start an appliance (formerly "boot").

stream selector

A filter for identifying an entity for which you want to throttle access.

string map

A NetScaler entity, consisting of key-value pairs, that can be used for pattern matching in all NetScaler features that use the default policy syntax.

striped configuration

A configuration available on all nodes of a cluster.

subnet IP (SNIP)

Subnet IP address. A NetScaler-owned IP address used for server-side connections.

SureConnect

A NetScaler feature that directs requests to an alternative web page if the primary page is DOWN.

Sysid

See system ID

SYSLOG

A standard logging protocol, implemented on a SYSLOG auditing module, (which runs on the monitored appliance), and a SYSLOG server, which can run on a remote system. SYSLOG uses User Data Protocol (UDP) for data transfer.

system ID (sysid)

A number, or possibly characters, identifying an appliance or virtual appliance.

T

thick provisioned format

VMDK format in which all physical disk space required for a virtual disk is allocated and zeroed out (wiped) when the disk is first created. In other words, space for a thick provisioned VMDK is reserved in advance for that VMDK only. You cannot overallocate physical disk space if you use thick provisioned VMDKs, which limits the number and size of VMDKs and can waste physical disk space, but ensures that you will have enough physical disk space in all circumstances.

thin provisioned format

VMDK format in which physical disk space needed for a virtual disk is allocated and zeroed out (wiped) only when the VMDK is written to the physical disk. In other words, space for a thin provisioned VMDK is allocated dynamically as needed; physical disk space is not allocated and reserved in advance. You can overallocate physical disk space if you use thin provisioned VMDKs, which allows you to put more and larger VMDKs on a given physical disk and avoid wasting unused space, but risks running out of physical disk space if your VMDKs are too full.

time stamp

Data indicating when an event occurred.

timeout

A setting indicating when an entity is to become unavailable (for example, how long a connection can remain idle without being closed). Also, the act of becoming unavailable after the specified period of time.

tracing

The use of trace files to debug problems in the flow of packets to the cluster nodes. The NetScaler operating system includes a utility called nstrace, which provides a dump of the packets received and sent by the appliance, and stores the packets in trace files. You use the Wireshark application to view the trace files.

traffic domains

A NetScaler feature with which you can create multiple isolated environments within a the appliance. An application belonging to a specific traffic domain communicates with entities and processes traffic within that domain. You can, for example, use the same IP address in different domains.

transparent mode

An operational mode in which an appliance between the end points of a connection does not have its own IP address. The appliance intercepts packets that one end point sends to the other.

two-arm mode

A deployment mode in which two network interfaces on the deployed appliance are connected to different Ethernet segments.

U

Use Source IP (USIP)

A NetScaler mode in which the ADC uses the client's IP address, instead of a SNIP address, in packets sent to the server.

USNIP

NetScaler mode that uses a subnet IP (SNIP) address as the source IP address of packets sent to the server, and as the address at which packets are received from the server. This mode is enabled by default.

V

view based access control model (VACM)

An SNMPv3 feature that enables you to configure access rights to a specific subtree of the MIB on the basis of various parameters, such as security level, security model, user name, and view type. You can configure agents to provide different levels of MIB access to different managers.

virtual IP (VIP)

A virtual IP (VIP) address is the IP address associated with a virtual server. It is the IP address to which clients connect for access to one of the servers represented by the virtual server. An appliance managing a wide range of traffic might have many virtual servers, each configured with its own VIP address. Some of the attributes of a VIP address are customized to meet the requirements of the virtual server.

virtual machine disk (VMDK)

File format for virtual disk drives. Originally developed by VMWare, but now an open format that is widely used in many types of clouds and virtual machines (VMs).

VMDK

See virtual machine disk

virtual server

A NetScaler entity with an IP address to which clients send requests. Distributes the requests to physical servers.

VPX

See NetScaler VPX.

W

waterfall chart

A NetScaler Insight Center chart that shows the cumulative effect of sequentially introduced positive or negative values.

Web 2.0 push

A NetScaler feature in which the NetScaler ADC functions as a proxy server to offload long-lived client TCP connections

and maintain relatively fewer, reusable connections to the server.

Web Insight

A component of NetScaler Insight Center. Monitors HTTP traffic.

Web Interface

The XenApp/XenDesktop component with which users access their applications, content, and desktops through a web browser. Also supports access through the Citrix XenApp plug-in. A NetScaler ADC can manage Web Interface access to XenApp or XenDesktop server farms.

web server logging

A NetScaler feature that sends logs of HTTP and HTTPS requests to a client system for storage and retrieval.

wildcard virtual server

A virtual server that accepts all traffic.

X

XenApp

A Citrix on-demand application delivery solution that enables any Windows application to be virtualized, centralized, and managed in the datacenter, and instantly delivered as a service to users anywhere on any device.

XenCenter

Management application for XenServer. You can use XenCenter to create, deploy, manage, and monitor virtual machines (VMs) from a Windows computer.

XenDesktop

A Citrix desktop virtualization and VDI solution that delivers a complete Windows desktop experience as an on-demand service to any user, anywhere.

XenServer

The Citrix open-source virtualization platform.